DOCUMENT RESUME

ED 160 040 IR 005 876

AUTHOR	White, Lee J.; And Others
TITLE	A Sequential Method for Automatic Document Classification.
INSTITUTION	Ohio State Univ., Columbus. Computer and Information Science Research Center.
SPONS AGENCY	National Science Foundation, Washington, D.C. Div. of Science Information.
REPORT NO	OSU-CISRC-TR-75-5
PUB DATE	Aug 75
GRANT	GN-36340
NOTE	140p.; For related document, see IR 005 877
EDRS PRICE	MF-$0.83 HC-$7.35 Plus Postage.
DESCRIPTORS	Algorithms; *Automatic Indexing; Bayesian Statistics; *Classification; Cluster Grouping; Data Bases; *Documentation; Flow Charts; Mathematical Models; Probability; *Sequential Approach; Statistical Analysis
IDENTIFIERS	*Keywords

ABSTRACT

The major advantage of sequential classification, a technique for automatically classifying documents into previously selected categories, is that the entire document need not be processed before it is classified. This method assumes the availability of a priori categories, a selection of keywords representative of these categories, and the a priori probabilities of the keywords within each category. In practice, these categories and keyword probabilities are constructed from a randomly selected document sample set. The performance of the sequential technique has been evaluated by classifying diverse data bases. The sequential technique was compared directly with the Williams' discriminant analysis method, and its performance compared very favorably. A series of experiments was also conducted which involved a full-text data base and a hierarchical data base of epilepsy abstracts. A technique has been developed for detection of those keywords which are "noisy" and adversely affect classification. This technique, called the Bayesian distance criterion, is also useful for obtaining multiple classes associated with a document. It is anticipated that the results of this research will find application in the classification and retrieval of library documents and in interactive document retrieval. (Author?CMV)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED160040

(OSU-CISRC-TR-75-5)

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

A Sequential Method for Automatic Document Classification

by

Lee J. White, Joseph D. Smith,
Gautam Kar, Doyle E. Westbrook,
Barry J. Brinkman, and
Robert A. Fisher

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Celianna Taylor

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Work performed under Grant No. GN-36340,
Office of Science Information Service,
National Science Foundation

The Computer and Information Science Research Center
The Ohio State University
Columbus, Ohio 43210

August 1975

IR 005 876

PREFACE AND ACKNOWLEDGMENTS

This report describes a study supported by Grant GN-36340 from the Office of Science Information Service of the National Science Foundation. This research, performed during the period December 1, 1972 through May 31, 1975, has considered the feasibility and applications of a sequential technique for automatic document classification.

This study is published by the Computer and Information Science Research Center of The Ohio State University as OSU-CISRC-TR-75-5 of its Technical Report Series. The Center is an interdisciplinary research organization which consists of the staff, graduate students, and faculty of many University departments and laboratories. The research was administered and monitored by The Ohio State University Research Foundation.

The authors would like to thank Dr. James E. Rush for his original ideas and contributions to this research. Other associates who have contributed to this research during the period of effort are: Sally E. James, George P. Jones, and Bryan W. Schurle.

We also wish to acknowledge those individuals who supplied the data bases used throughout this study. We would like to thank Robert Ruberti of the Rome Air Development Center who made available to us copies of Williams' computer programs and the solid state data base. In addition, we would like to thank Rita Lerner of the American Institute of Physics, who provided us with the Searchable Physics Information Notices data base, and Dr. Keffin Penry of the National Institute of Neurological Diseases and Stroke, who provided us with the Epilepsy data base.

ABSTRACT

Sequential classification is a technique for automatically classifying documents into previously selected categories. The major advantage of this method is that the entire document need not be processed before it is classified. The sequential method assumes the availability of a priori categories, a selection of keywords representative of these categories, and the a priori probabilities of the keywords within each category. In practice, these categories and keyword probabilities are constructed from a randomly selected document sample set.

The performance of the sequential technique has been evaluated by classifying diverse data bases. The most extensive tests have involved abstracts from the Searchable Physics Information Notices (SPIN) data base. The sequential technique was compared directly with the Williams' discriminant analysis method, and its performance compared very favorably with this well known approach. In addition, a series of experiments was conducted which involved a full-text data base (Kucera-Francis) and a hierarchical data base of Epilepsy abstracts.

A technique has been developed for detection of those keywords which are "noisy" and adversely affect classification. This technique, called the Bayesian distance criterion, is also useful for obtaining multiple classes associated with a document. It is anticipated that the results of this research will find application in the classification and retrieval of library documents and in interactive document retrieval.

TABLE OF CONTENTS

		PAGE
Chapter 1	Introduction	1
Chapter 2	Automatic Document Classification and Keyword Selection	8
	2.1 Statistical Methods of Automatic Document Classification	9
	2.2 Clustering Techniques	11
	2.3 Keyword Selection	15
Chapter 3	The Sequential Classification Method	21
	3.1 The Sample and Test Set of Documents	22
	3.2 Conditions for Evaluation of the Sequential Algorithm	23
	3.3 Keyword Frequency	24
	3.4 Hash Tables	25
	3.5 The Sequential α-Test	26
	3.6 Evaluation Criteria for Experiments	31
Chapter 4	Experiments with the Sequential Algorithm Using the SPIN Data Base.	33
	4.1 Classification Experiments	35
	4.2 Experimentation With the Sequential Algorithm	42
	4.3 Keyword Set Experiments	46
Chapter 5	Comparison of the Sequential and Discriminant Analysis Classification Techniques	58
	5.1 Discriminant Analysis Approach	58
	5.2 Solid State Experiments	59
	5.3 SPIN Experiments	62
	5.4 Conclusions	62
Chapter 6	Sequential Classification Experiments Using the Kucera-Francis and Epilepsy Data Bases	65
Part I	The Kucera-Francis Experiments	65
	6.1 Data Base Classification and Keyword Selection	65
	6.2 Sequential Classification Experiments	66
	6.3 Summary of Kucera-Francis Experiments	70
Part II	Experiments with the Epilepsy Data Base	72
	6.4 Overview of the Epilepsy Data Base	73
	6.5 Selection of the Sample and Keyword Sets	75
	6.6 Sequential Classification Experiments	75
	6.7 Summary of Epilepsy Experiments	79
Chapter 7	Overview of the Sequential Experiments	80
	7.1 Selection of Keywords	80
	7.2 Experimentation with Parameter Values	81
	7.3 Experimentation with the Classification Algorithm	82
	7.4 Elements of Sequential Classification	83
	7.5 Conclusions	84

Chapter 8 The Bayesian Distance 85

8.1 Limitations of the Sequential Method 85
8.2 Vector Representation of Keywords 88
8.3 Vector Space Representation of Documents 88
8.4 Definition of the Bayesian Distance 90
8.5 Variation of Bayesian Distance with Keywords. 91
8.6 Analysis of Ideal Documents 92
8.7 Analysis of Non-Ideal Documents 97

Chapter 9 The Revised Sequential Method 102

9.1 Separation of Good and Noisy Keywords 102
9.2 The Bayesian Distance Calculator and Noise Detector 105
9.3 Description of the Revised Sequential Algorithm 107
9.4 Implementation of the Algorithm and Results of Experiments 111
9.5 Secondary Classification 115

Chapter 10 Conclusions and Future Work 116

10.1 Conclusions of this Study 116
10.2 Future Work 118

Appendices

A Keyword Set Generation for SPIN - 11 Classes A-1

B Description of Rocchio's Algorithm B-1

C Measure of Comparison of Two Classifications C-1

D Operational Characteristics of the Sequential Algorithm D-1

E Keyword Hashing Algorithm E-1

References

CHAPTER 1

INTRODUCTION

With the increasing use of optical character recognition equipment, direct teletype input and other forms of high-speed, high-volume input devices for computers, the processing of narrative abstracts and documents is becoming more practical. Methods are required for automatic indexing and abstracting, and for subsequent dissemination or extraction of important information. The research described in this report examines the feasibility of automatically classifying a document data base by a sequential technique. The ultimate purpose is to facilitate the interactive retrieval of those documents desired by the user from the data base.

A number of algorithms for classification of documents by computer have been proposed for use in information systems in which large collections of documents are involved. Both statistical and nonstatistical algorithms have been considered and have been implemented and tested on small data bases. Early efforts by Maron [27] and also by Borko and Bernick [5] showed the potential of this approach, but also identified a number of problems to be resolved. A later effort by Williams used discriminant analysis [42,43], and led to results of fairly high accuracy in the 60-80% range. Because quantitative results were reported by Williams, and his computer programs and data bases were made available, comparative studies were conducted. These are described in Chapter 5 of this report.

The requirements of processing time and computer memory, both core and auxiliary storage, are serious problems for all automatic classifications algorithms when these techniques are applied to large data bases. Classifications based upon linear-algebraic techniques have large processing time

and storage requirements although high classification accuracy can be achieved. Classification obtained without processing entire documents is desirable. A further desirable feature is that the classification procedure be essentially independent of the document collection, i.e., can be flexibly adapted to different or dynamic data bases.

The research described in this report is a feasibility study of a sequential analysis technique for automatically classifying documents. The sequential method has low processing time and storage requirements and does not require whole-document analysis. The approach is very flexible, and can be adapted to data bases of various content or description. The object of the feasibility study was to ascertain whether such a simplified approach could still obtain a high classification accuracy when applied to diverse types of document collections.

This sequential classification approach is based upon the notions of sequential analysis as described by Wald [41], which have been successfully applied in the field of pattern recognition. For example, Fu has applied these ideas to the problem of character recognition [17]. Fried and his co-workers originally suggested that this approach be applied to automatic classification in 1968 [16]. An initial implementation of this idea was provided by Aberi [1], and the current research effort has contributed substantial numerous improvements in this implementation.

The sequential classification method assumes the availability of K categories, a selection of keywords representative of these categories, and the a priori probabilities of the keywords within each category, denoted $P(X_i|C_j)$, where X_i indicates the i-th keyword and C_j the j-th category. Also, the a priori probability of each class, which is either previously available or derived, is also taken into account, and is denoted $P(C_j)$. Given the K categories, the keywords and their probabilities are usually

determined from a representative sample set of documents from the data base. A discussion of the use of this sample document set, as well as the extraction of keywords, is given in Chapter 2.

The sequential method of automatic classification differs in a number of ways from other methods reported in the literature, most of which involve matrix manipulations (for example, similarity matrices - Salton [32,34], or discriminant analysis - Williams [42,43]). A major advantage of the sequential approach over previously proposed methods is that it is not necessary to read an entire document before a decision about its classification can be made. This fact permits the realization of substantial savings in processing time. Keywords and stoplist words are kept in hash tables which further improve the performance of the model. In spite of these tables, large savings in storage requirements are still realized when compared with methods using matrix techniques, particularly for large data bases. Another advantage of the sequential technique is the variety and flexibility of the prediction methods that can be used. Various parameters may be changed for successive runs or within a single run as desired or demanded by the data being processed. Furthermore, using the α-prediction rule, documents are classified within a certain degree of accuracy. In most other automatic classification methods, an implicit assumption is made that either a document belongs to a given category or it does not. A more detailed description of the sequential classification method and its properties will be given in Chapter 3.

Numerous tests of the performance of the sequential classification technique have been conducted using diverse data bases. The performance measures that have been investigated are classification accuracy, speed, and core size requirements. Most extensive tests have been made using physics abstracts made available by the American Institute of Physics in

the form of several tapes of the "Searchable Physics Information Notices", referred to as the SPIN data base [35]. Approximately 4,000 abstracts are now available which average about 150 words in length. A reliable hierarchical classification was provided by manual indexing, but keywords were not available. A major keyword extraction effort has resulted in those keywords used in the experiments described in detail in Chapter 4. These experiments indicate that from 75 to 87% accuracy has been achieved using different parameter values with the sequential method, which is comparable to the best results reported in the open literature [43].

The variation of the algorithm parameters has been shown to affect accuracy and computation time, but the algorithm is not unstable and is not subject to great variations in performance with small changes in parameter values. The greatest advantage of this method is the speed with which classification can be achieved. For 500 documents of the SPIN data base, the required execution time to classify these documents was 53 seconds when entire documents were read, whereas the same program required only 17 seconds using the sequential method of partial document processing.

Some of the highest classification accuracies reported in the literature were due to the discriminant analysis method of Williams [42, 43]. Thus, an effort was made to acquire both the computer programs and data bases used by Williams in order to compare it to the sequential classification method. In all, 1743 solid state abstracts were obtained which he used in his reported research. Of these, classification of 420 are known in terms of three given classes, and the keyword sets used were also provided.

Three keyword sets and two document samples from the solid state abstracts data base were tested using both Williams' discriminant analysis and the sequential analysis methods. The accuracy of the sequential analysis

method compared very favorably to that of discriminant analysis, and these experiments are described in detail in Chapter 5. When these two methods were compared by an experiment using the SPIN documents with a small keyword set, the results actually show the sequential method to be superior. This experiment and the results are also reported in Chapter 5. The discriminant analysis method requires a number of matrix manipulations to find the discriminant characteristics for each class. Although this is a one-time calculation once the sample has been chosen, it is a very costly procedure in terms of execution time. Sequential analysis requires no such setup costs. Further, keywords or new sample documents can be added or changed at little cost. Sequential analysis is slightly faster in execution than discriminant analysis, and does not have tight limits on the number of keywords, classes, and documents which describe the data base as is the case with matrix techniques. In sequential analysis, the execution time and storage requirements increase with the number of keywords and classes, but only slightly relative to the time and storage increases with discriminant analysis.

Chapter 6 describes the other two data bases used in this study and the experiments performed on them. The largest and most diverse data base available is a one million word data base provided by Kucera and Francis at Brown University [25]. This consists of 500 documents in widely different subject areas, but category membership is not noted in the data base itself. Each document consists of about 2,000 words, since in every case the document was discontinued at the end of the first full sentence after the two-thousandth word. A major indexing effort has produced appropriate categories, key words, and manual document classification. An assistance in this effort has been a compilation of word frequency for all words occurring in the data base and published by Kucera-Francis [25].

Over 1,200 abstracts were obtained from the Epilepsy Section of the National Institute of Neurological Diseases. The primary reason for their acquisition was that both keywords and a hierarchical classification are provided, and other workers in the field have reported good results using this data base. Experiments on this data base will also be reported in Chapter 6.

An overview of the results of the application of the sequential classification method to these four data bases is given in Chapter 7. Some general inferences are drawn concerning the use of the sequential approach with an arbitrary data base.

Recall that in the sequential approach, only as much of each document is read until it can be unambiguously classified into one or more classes or subclasses. Despite the obvious advantages of this approach, it may be susceptible to several keywords occurring near the beginning of the document which are either incorrect indicators of the content of that document and not associated with the appropriate class or classes, or else indicate a different aspect of the document. A method has been devised which is very sensitive to such inappropriate indicators and this is the Bayes distance criterion (denoted D_B). This measure is related to probability of error at each stage and also to the uncertainty or entropy, but the main point here is that D_B is extremely sensitive to an unexpected keyword. For example, in an ideal document in which all initial keywords are indicative of a single class, the measure D_B is found to rise monotonically and rapidly to a unity value. For a document containing a keyword not consistent with that class, there is an immediate observable drop in D_B. These relationships are described in greater detail in Chapter 8, together with some experimental verification of these ideas given in Appendix E.

A modified sequential method is constructed using this concept of Bayes distance to identify "noisy" keywords, and is described in Chapter 9. Instead of using each keyword directly as it is read from the document to be classified, the keywords instead are entered into a "useful" buffer or into a "noise" buffer depending upon its effect upon the Bayes distance. Experiments show this not only improves classification accuracy for a single class situation, but also represents a more systematic approach to multi-category classification.

In Chapter 10, some indications of future work will be given, along with conclusions reached with the completion of the reported research.

CHAPTER 2

AUTOMATIC DOCUMENT CLASSIFICATION AND KEYWORD SELECTION

Many methods exist for ordering or classifying the elements of a data base into homogeneous groups based on similarities in the attributes of the elements. In the area of information storage and retrieval, the elements are usually documents and the attributes are words or concepts characterizing the documents. In the first part of this chapter we will review some of these existing methods and attempt to point out their limitations. In the second part, the importance of choosing the right attributes for classification will be emphasized and since for documents the attributes are usually keywords, some methods of keyword selection will be presented.

Techniques for classification of document collections may be divided into two basic categories:

a) an a priori classification of a given document exists, and each document is placed into the cluster whose centroid is most similar to that document, and

b) no a priori classification is specified, and clusters are formed only on the basis of similarities among documents.

Classification schemes of the first type are very common and often involve manual methods. For example, new acquisitions of a library are classified by placing them into the clusters of a standard a priori classification. Problems of the second type are more difficult to handle and automatic or semi-automatic methods are often used.

2.1 Statistical Methods of Automatic Document Classification

The problem of automatic document classification has been an area of research for a long time. One of the earliest and most significant contribution was that of Luhn [26] who showed that a statistical analysis of the words in a document can provide some clues as to its content. This work led to the development of several automatic classification methods from the early 1960's to the early 1970's. The basic approach of all these methods is the same. Each document that is to be classified into one of the given categories is read entirely and all the keywords present in it are extracted. Using these keywords a prediction function is used to relate the document to each of the categories. The differences in the various methods lie in the nature of the prediction function that is used and the way the relationship between a category and a document is computed. These differences and similarities will be clarified when the various methods are discussed in some detail in the following sections.

2.1.1 Bayesian Technique

Maron [27] applied a statistical method to the problem of automatic classification which involved:

i) the determination of certain probability relationships between individual keywords and subject categories, and

ii) the use of these relationships to predict the category to which a document belongs by using Bayes rule.

The classification techniques to be described in this research bear resemblance to Maron's method in that the keyword class frequencies are calculated using sample documents and the a posteriori probabilities of the classes after a number of keywords are read and computed using Bayes rule. The basic difference lies in that this research hypothesizes that the keywords present in a document need not be examined in toto before class membership can be determined, but

need only be read sequentially until a decision is reached. The philosophy behind such a sequential technique will be explicated in greater detail in Chapter 3.

2.1.2 Factor Analysis

Subsequent to the publication of Maron's work, Borko [5,6,7] devised a technique for automatic document classification based upon factor analysis. Counts were made of the number of times each of the keywords occurred in the document sample set. A keyword correlation matrix was then defined, factor analyzed, and a resulting set of subject categories were identified. A prediction technique based upon the keyword frequencies in the documents and their factor loadings was developed.

2.1.3 Discriminant Analysis

At about the same time as Borko, Williams [42,43] devised a classification technique based on discriminant analysis. Instead of computing factor loadings, he computed a discriminant coefficient for each keyword and category. Using a set of documents already manually classified, the frequency of occurrence of a keyword in each document belonging to a given category is then empirically obtained. Using these frequencies, two m x m matrices are computed as follows:

W ≡ a matrix whose elements are the pooled within-category sum of squares, and

A ≡ a matrix whose elements are the among-category sum of squares.

If a document contains n keywords, whose mean frequencies are given by $x_1, x_2, \ldots, x_n$, then a prediction function of the following form is used:

$$X = B_1x_1 + B_2x_2 + \ldots + B_nx_n \qquad (2\text{-}1)$$

where the B_i coefficients are obtained by solving the equation

$$|W^{-1} A - \lambda I| = 0. \qquad (2\text{-}2)$$

One of the eigenvalues λ is chosen to give the best discrimination between

the categories. The eigenvector corresponding to this λ then provides the set of coefficients $\{C_i\}$. The discriminant score obtained from equation (2-1) is then the basis of assigning a test document to one of the categories. Chapter 5 will further discuss this approach, as the discriminant analysis method of Williams and the sequential approach are compared.

Both the methods of Borko and Williams suffer from the same disadvantage. They require the inversion of matrices to compute the coefficients which relate a keyword to a category. For a sizable set of keywords, say 500, these methods become impracticable. More specifically, the increase in storage and time required by these methods is proportional to the square of the number of keywords.

2.1.4 The SMART Retrieval System

The SMART document retrieval system was designed by Salton [32,33,34] in the early sixties, and presently offers an experimental facility for further research. The system takes documents and search requests, performs an automatic content analysis of the texts, matches analyzed documents with search requests, and retrieves those documents believed to be most similar to the user submitted queries. The system is quite complex, and consists of a large number of interacting routines and subsystems, but for this development we would like to concentrate on several of the clustering techniques designed for use with the SMART system. One of these clustering methods is due to Dattola [9] and another developed by Rocchio [29]. These will be discussed in the next section as they were used for the purpose of classification and keyword analysis in the present study.

2.2 Clustering Techniques

The theory of clustering deals with the problem of finding natural groupings in a set of data. These natural groupings are obtained based on the similarity in the attributes of the data elements. In this section we will discuss the basic philosophy of clustering and show how it has been applied for use in automatic document classification.

2.2.1 The Similarity Matrix

The starting point for most clustering algorithms is the similarity matrix. Let $D = \{d_1, d_2, \ldots, d_t\}$ be the set of documents under consideration. Each document d_i in D is read entirely and all keywords occurring within it are extracted. A t x t matrix Q is obtained such that

Q_{ij} = number of keywords occurring in common between documents d_i and d_j.

The similarity matrix S can now be directly obtained from the matrix Q by normalizing each of its elements by using a standard similarity measure. There are several such measures that have been used in the literature and the one to be used for a particular implementation depends on the user requirements. One that has been used very widely is the Tanimoto similarity measure [32], by which each element of S is calculated:

$$S_{ij} = \frac{Q_{ij}}{Q_{ii} + Q_{jj} - Q_{ij}} . \qquad (2\text{-}3)$$

As can be seen, this similarity matrix is symmetric and the diagonal entries are all unity. Therefore, during use in clustering only a triangular portion of the matrix needs to be stored.

A set of clusters is obtained using this similarity matrix. This is usually done by defining a threshold parameter such that two documents d_i and d_j are assigned to the same cluster if S_{ij} exceeds this parameter. It is clear that as this threshold is increased two documents must have a greater number of keywords in common in order to be placed into the same cluster.

Most clustering methods follow this general procedure. The next section briefly examines several such methods.

2.2.2 Document Clustering by Finding Centroids

With this approach, every document is examined serially and some documents are chosen to be centroids of the clusters if they satisfy certain criteria.

For example, Rocchio's method [29] determines whether a document should be the centroid of a cluster by testing if there are sufficiently many documents located in its vicinity. Specifically, a document is a possible centroid only if there are at least N_1 documents which have a similarity of at least p_1, and N_2 documents which have a similarity of at least p_2 with the given document where N_1, N_2, p_1, and p_2 are parameters which can be varied to obtain a different number of clusters. This approach ensures that documents on the "edge" of large clusters cannot in turn become the center of another cluster.

Dattola's method [9] further simplifies this process by dividing the total document collection into an arbitrary number of groups. Each group is represented by a centroid whose dimension is equal to the number of keywords in the system. Each element of this vector is a quantity proportional to the frequency of occurrence of a keyword in all documents of the given group. Every document in the collection, each of which is also represented by such a vector, is now compared with each centroid and reassigned to the group to which it is the closest. The process is repeated until identical sets of clusters are obtained in successive iterations.

The importance of these clustering algorithms in this study are for keyword investigation and for obtaining natural clusters or classes. For most of the investigations of the SPIN documents reported in Chapter 4, Rocchio's method was found more effective than that of Dattola, though both were utilized in this study. For this reason, the details of Rocchio's algorithm are documented in Appendix B.

2.2.3 Comparison of Two Classifications

In Chapter 4, it will be required to be able to compare two classifications, and to determine a measure of their closeness. In this section such a measure is proposed which is applied in Chapter 4 and an optimality property of this measure will be proven in Appendix C.

For the purpose of this discussion, we will define a classification K of a set of objects (documents) D to be given as a set of categories, each of which is a subset of the objects. In order for this measure to be as general as possible, assume that an object may belong to more than one category, although considerable simplification of the measure will result if the set of categories just comprise a partition of the set D.

Given two classifications, K_1 and K_2, of the same set D, we consider a measure of the distance, or difference, between them, denoted $||K_1 - K_2||$. Let D consist of m objects, and K_1, K_2 are comprised of k_1 and k_2 categories respectively, with $k_1 \leq k_2$. Furthermore, since an object may belong to more than one category in either of the classifications, denote by $(t_j)_i$ the number of times the i^{th} object occurs in the j^{th} classification.

Define an elementary change of one classification to another as follows: an elementary change consists of either

a) the movement of the occurrence of an object from one category to either an existing category or a newly created empty category, removing its occurrence from the original category;

b) the deletion of the occurrence of an object in a category, as long as that object still occurs in some other category of the classification;

c) the creation of an occurrence of an object in a category, as long as that object does not already exist within that category.

Define the difference between two classifications, denoted $||K_1 - K_2||$, as the minimum number of elementary changes require to transform K_1 into K_2 taken over all possible ways of making that transformation by sequences of elementary changes. The number of such possible sequences can be kept finite by not creating new elements indefinitely.

This difference $||K_1 - K_2||$ is proposed as a measure of the difference between two classifications. However, this measure will not only depend upon the difference between classifications, but also will tend to grow with the number of objects in D, i.e., parameter m. A further complication are the counts $(t_j)_i$, when all these counts are not simply unity. Thus, in order to normalize this measure, rewrite the final measure as $||K_1 - K_2||/T$, where

$$T = \sum_{i=1}^{m} \text{Max}[(t_1)_i, (t_2)_i].$$

There remains the problem of how this measure can be computed, as it is defined utilizing an optimality condition. Although there are a finite number of sequences of elementary changes to investigate, it would be prohibitively time-consuming to do so. Using the well-known assignment algorithm, which is quite efficient, we can apply it as follows. Define a $k_1 \times k_2$ matrix B as

$$b_{ij} = |C_i \cap C_j|,$$

where vertical bars denote set cardinality, and C_i is a category from classification K_1, C_j a category from classification K_2. The following theorem relates matrix B and the assignment algorithm to our measure of classification difference.

Theorem

If k is the maximum assignment sum for the matrix B, then

$$||K_1 - K_2|| = \{\sum_{i=1}^{m} \text{Max}[(t_1)_i, (t_2)_i]\} - k.$$

The proof of this theorem will be given in Appendix C. The measure $||K_1 - K_2||/T$ will be applied in Chapter 4.

2.3 Keyword Selection

Keyword selection is a process of analyzing a set of words for the purpose of selecting a subset which is suitable for distinguishing between given document

classes. The keyword concept is based on the assumption that in a given collection of documents, the basic information needed for classifying a document is summarized in just a few keywords. Most of the automatic keyword selection methods that have been described in the literature or have been used in actual systems have been statistical in nature. That is, the value of a keyword is considered to be directly proportional to its frequency of occurrence in the document collection.

The sequential method of document classification uses the relative frequencies of keywords to discriminate between classes. Thus, good keywords should have a high frequency of occurrence within a class that they describe best, and have a low frequency of occurrence within other classes. There is an interesting trade-off concerning the frequency of a keyword; if a keyword occurs too frequently, it will tend to point to too many classes for adequate discrimination. On the other hand, even if every occurrence is associated with a single class, a keyword with too low a frequency will not occur frequently enough to be useful within new documents to be classified.

A number of keyword selection or modification techniques will be described in this section, and these will be applied to various data bases in Chapters 5 and 6, but will be most extensively used in Chapter 4 with the SPIN data base. First, keyword selection techniques will be outlined in general, a few specifics being presented in Appendix A. An automatic keyword generation method will be presented which is shown in Section 4.3.2 to be quite successful for sequential classification. Finally, a technique for iteratively improving keyword sets will be indicated, called the "good-bad" keyword criterion.

2.3.1 Manual Keyword Selection

The starting point for the semi-automatic keyword generation method is a stoplist of very commonly occurring words. A general stoplist of prepositions, articles, pronouns, etc., was available from previous work, but was modified manually and generally increased in size to 1,225 words.

Initially keywords were defined as any word not in the stoplist which occurred at least three times within the documents associated with the class in question. This proved to be unsatisfactory, as the resulting keywords failed to possess sufficient discriminating power for sequential classification. A systematic study of keyword selection resulted in the following general set of criteria:

a) the keyword should occur in at most t_1 of the t categories;

b) the keyword should occur at least t_2 times within the set of sample documents indicative of the desired class;

c) the keyword should be distributed in the sample documents in such a a way as being strongly indicative of the primary class and perhaps a secondary class.

Criterion a) ensures that the chosen keyword has sufficient discriminatory power, and together with other keywords present, will eliminate a sufficient number of classes by its presence in a document to be classified. Criterion b) assures the keyword occurs frequently enough to be useful for classification. Criterion c) depends upon the number of classes and number of keywords which can be utilized. The best example of an objective and quantitative application of these criteria is given for automatic keyword selection in Section 2.3.2.

For the manual selection of keywords described in Section 4.3.1 for the seven classes of SPIN, each keyword was required to occur in at least two different documents and at least three distinct occurrences counted within all the sample documents of the indicated class. In general, keywords should occur in documents corresponding to no more than three classes. A study of these distributions was made to select keywords indicative of one or perhaps two classes. In addition, some keywords of low frequency were chosen manually if they seemed indicative of the corresponding class.

2.3.2 Automatic Keyword Selection

In Chapter 4, several experiments are performed with the sequential algorithm on the SPIN data base using automatically generated keywords. These were very successful in that classification accuracies were obtained close to those using manually selected keywords.

For seven SPIN classes, the following automatic keyword selection criterion is investigated in Section 4.3.2:

1) each keyword should occur in at most t_1 categories;
2) each keyword should occur at least t_2 times within the sample documents of the corresponding class;
3) select any such keyword which occurs only within sample documents of the desired class;
4) if the keyword occurs within sample documents for only two classes, at least t_3% of the occurrences should be in one class of documents;
5) if the keyword occurs within sample documents for three classes, then at least t_4% of the occurrences should be in the primary class of documents, and the tertiary class corresponds to no more than t_5% of the occurrences.

This criterion is applied in Section 4.3.2, and a study has resulted in best choices for the parameters t_1 through t_5.

Extensive experiments on similar criteria were conducted for eleven class SPIN data. Keywords occurred within sample documents over at most either three or four classes. The criteria are given in detail in Appendix A, and are supportive of the experiments reported in Table 4.9 of Section 4.3.4. In the definition of these keyword sets, the criterion described in the next section is also applied.

2.3.3 "Good-Bad" Iterative Keyword Improvement Criterion

Even keyword sets which are systematically chosen and in general are appropriate for use with the sequential algorithm may contain specific keywords which lead to misclassification. It was discovered that many words which were intuitively "good" in describing a given class actually degraded performance, leading to substantial misclassification error.

A criterion was developed by which specific keywords could be evaluated in terms of their overall influence on document misclassification. As such keywords were identified and removed from the keyword set, the process could be iteratively applied, resulting in a monotonically decreasing keyword set which produced improved classification using the sequential algorithm and these modified keyword sets. This criterion was ad hoc, but the results reported in Chapters 4 and 6 show it to be effective. This led to a much more systematic method for identifying "noisy" keywords which is discussed in Chapters 8 and 9.

The ad hoc method to be described here was called the "good-bad" criterion. The value of a keyword in the classification of a particular document was defined as:

1) the keyword is "good", or aids performance if the probability of the keyword in any of the correct categories exceeds the probability of the keyword in every incorrect category by at least p_1;
2) the keyword is "bad", or degrades performance if the probability of the keyword in any of the incorrect categories exceeds the probability of the keyword in every correct category by at least p_2.

Those documents for which each keyword aided and degraded performance were summed over the documents classified. The difference between these counts is called the "good-bad" count. Clearly if a keyword indicated a correct choice more frequently than it indicated misclassification, then the keyword should be retained. The

problem keywords are detected when they lead to incorrect classification more frequently than they indicate the correct class. As these keywords are identified and removed from the keyword set, a subsequent classification of the same documents using this modified keyword set should result in improved classification.

Although this method is ad hoc, it did result in improved classification, as long as the original keyword sets were sufficiently large and appropriately selected. As the "good-bad" criterion was iteratively applied, the number of documents misclassified decreased, but a trade off was discovered as fewer keywords were used, the documents contained fewer keywords, and more documents had to be read entirely. Throughout this study, $p_1 = p_2 = 0.005$ was proven to be effective for the SPIN data base, but clearly these thresholds depend upon the relative frequencies encountered and the number of classes considered. Tables 4.6 and 4.9 show the effectiveness of this criterion, and to see exactly how it is applied and typical "good-bad" counts, see Appendix A.

CHAPTER 3

THE SEQUENTIAL CLASSIFICATION METHOD

This chapter contains a more detailed description of the sequential classification method, and Appendix D should be referenced for operational and computer program information. In this chapter various parameters of the sequential method are indicated together with their function and how they may be selected. A number of experiments testing the sequential algorithm will be described in subsequent chapters, while conditions and evaluation criteria will be discussed in this chapter for these experiments.

The problems encountered in a classification technique are: selection of the variables upon which the classification decision is to be based, computation of the classification statistics, determination of the decision rule, and analysis of effectiveness.

A user starts with a set of documents and decides on a group of subjects that interest him with respect to the contents of the document collection. He then constructs a tree of the classification structure desired for the total collection of documents under consideration. Each node of the tree represents a category, and all nodes or branches emanating from a single node are its subcategories. A category number is assigned to each node. Each document from a sample of the total collection is manually assigned to one or more nodes of the tree.

The variables used in the classification process are frequency statistics on word types from each category. Previous experiments have indicated that all word types do not need to be retained for the classification process, and computationally it would be impractical to do so. Ideally, words selected to represent the categories should occur in one and only one

category. However, usually only a few words in any data base occur in only one category, and these words do not necessarily occur in every document. Therefore the challenge is to utilize words which overlap into several categories, and to discern their function in this case by frequency counts.

In describing the sequential algorithm in this chapter, we assume that, for the document data base being considered, the classification structure and keyword sets are already available. If this is not the case, then some of the techniques described in Chapter 2 might be applied. However, in this chapter, a detailed discussion of the classification statistics and keyword frequency table preparation will be given.

3.1 The Sample and Test Set of Documents

The experiments described in Chapters 4 through 6 are designed to evaluate the sequential classification algorithm for document data bases of diverse content and description. The available documents are divided into two sets, called the sample and test documents. We assume that the sample set of documents are already preclassified. Keyword frequency tables by classes can then be prepared using the sample set; in some cases, the sample set might have to be used to define the keywords. Evaluation of the efficacy of the sequential algorithm then depends principally upon its application to the test set of documents, since higher classification accuracy would be expected to be achieved with the sample set than with the test set. In a typical application, the test documents would comprise the greater portion of the data base, and would not be preclassified since the purpose of the classification algorithm would be to achieve this result. For evaluation purposes, however, the test documents

are preclassified in order that the sequential classification algorithm can be evaluated. This approach is consistent with common practice in pattern recognition and other scientific areas, in which the sample document set might be viewed as "training samples", and then evaluation performed using test samples.

The size of the sample set is an important parameter to be determined. In the case of a large data base, the sample set would be chosen to be as large as practicable. Reliable classification of the sample is required. The most important factor in the selection of the sample is that it be representative in terms of the entire data base. Each class to be considered should be adequately represented in terms of documents so that a sufficiently representative keyword frequency count can be obtained. For the experiments described in subsequent chapters, the sample set was chosen large enough so as to assure sufficient representation of the class of lowest a priori probability, and yet small enough that an appropriate evaluation of the sequential algorithm can be made using the remaining test documents. The sample set was not allowed to exceed 30% of the documents available for evaluation. This decision was based upon a study made by Williams [43].

3.2 Conditions for Evaluation of the Sequential Algorithm

In order to evaluate the sequential classification algorithm, a number of conditions were imposed so that the test of the algorithm would be as severe as possible. The conditions considered were the following:

(1) The keywords were assumed to be independent, and a priori probabilities were obtained from frequency counts independent of neighboring keywords, their order, or the context in which the keywords are found. Any model which takes any of these dependencies into account could be

incorporated into the sequential algorithm, and any such change should improve the performance of the sequential algorithm over that reported here.

(2) As described below, as more keywords are read from the document, less likely classes are dropped from consideration. If all classes were kept until a decision is made, a more accurate decision would result but the sequential method is so designed to drop classes for reasons of efficiency. This feature is especially important in cases where many classes exist.

(3) If the entire document must be read before enough keywords are found for classification, the document is not treated as unclassifiable; instead, a classification decision is made at that point (see Section 3.6). Especially in the case of documents with an inadequate number of keywords, a higher classification accuracy would be obtained if these documents were deleted from the test.

(4) Studies have shown that keywords are not distributed uniformly throughout the document, but typically the most useful keywords are concentrated at the beginning and end of the document [43]. The tests conducted on the sequential classification method have read the document from beginning to end, although in general higher classification accuracy might be obtained by skipping to concluding portions of the document after reading the initial part, and then analyzing intermediate portions of the document.

3.3 Keyword Frequency

Next consider the preparation of the keyword frequency table and the a priori class probabilities which are used in the sequential α-test. Let D_j represent that subset of the sample documents associated with class C_j, and t_{ik} denote the number of occurrences of keyword W_i in document d_k.

Then the frequency of keyword W_i given that a document is in class C_j is

$$f(W_i|C_j) = \sum_{d_k \epsilon D_j} t_{ik}. \qquad (3\text{-}1)$$

These frequencies are calculated for each keyword and stored in a frequency table. When the keyword a priori probability estimates are used in the α-test, they are normalized by calculating the following:

$$P(W_i|C_j) = \frac{f(W_i|C_j)}{\sum_{k=1}^{N} f(W_k|C_j)} , \qquad (3\text{-}2)$$

where N indicates the number of keywords.

During classification, the keyword frequencies do not change, and thus the denominator in equation (3-2) is calculated only once. The a priori class probabilities, denoted q_j, are calculated as

$$q_j = \frac{\sum_{i=1}^{N} f(W_i|C_j)}{\sum_{k \epsilon L} [\sum_{i=1}^{N} f(W_i|C_k)]} , \quad \text{for each } j \epsilon L \qquad (3\text{-}3)$$

where L denotes the set of indices of classes remaining. As indicated previously for equation (3-2), the bracketed portion of the denominator of (3-3) does not change as document classification progresses. Yet L may become smaller as the α-test requires classes to be dropped from consideration. Thus the denominator of (3-3) will have to be recalculated, and q_j for remaining classes must be correspondingly updated.

3.4 Hash Tables

The keyword set and stop list were stored as hash tables, with keyword frequency information also available in the keyword hash table. The primary advantage of a hash coding or scatter storage method is that searches can be achieved by examining as few as one data cell of the table, or only slightly over this on the average. This method appears to be

ideally suited for the sequential classification implementation, as an extensive amount of searching is required. Tests showed that processing time was reduced by several magnitudes by using a hashing method to search keyword set and stop list over a sequential search, but that a binary search only resulted in marginal improvement.

The size of the hash tables was designed to provide a 60-80% loading factor, which yielded a low number of probes of the tables and resulted in few collisions. A systematic investigation of about twenty hashing algorithms showed that one of the simplest studied yielded the fewest collisions for both the stop list and keyword set. The hashing algorithm used is discussed in Appendix E.

The function of the stop list is to improve the processing of non-keywords which occur very frequently, and so thus it is especially important that the hashing algorithm for the stop list be extremely fast. Using the SPIN data base described in Chapter 4, fewer than 30% collisions were observed. When collisions did occur for the stop list words, Day's algorithm [10] only required an average of two address probes. The statistical performance of this approach is even better than this, however, since very high frequency words such as the, and, etc., are found in the stop list hash table with the first probe.

3.5 The Sequential α-Test

We are now prepared to give a detailed discussion of how the sequential classification algorithm is applied to a document to be classified, and especially how the α-test is performed. A document is read into a buffer,

or if the document is too large, a portion of it is read into the buffer. The title, if available, is also included, as experiments have shown it to be rich in keywords. Words are read sequentially from the document.

A word is read from the buffer, and is hashed into the stop list table. If it occurs there, the next word is read from the document. If a keyword is detected, then the keyword frequencies are retrieved from the hash table, and keyword probabilities calculated according to equation (3-2). Whether or not the α-test is performed depends upon two input parameters, T and R. T is the number of initial keywords that must be read before the first α-test is conducted. R denotes the number of keywords to be read between subsequent α-tests. Both parameters T and R prevent a precipitous decision from being made, especially in the presence of misleading or "noisy" keywords which happen to occur near the beginning of the document. When T=R=1, this means the α-test is conducted after each keyword is read. The function of parameters T and R are indicated in the sequential algorithm flow chart in Figure 3.1.

If the T and R conditions are satisfied, and the α-test is to be performed, using probabilities from equation (3-2), the a posteriori probabilities of each class are calculated as

$$\alpha_j = \frac{P(W_{i_1} W_{i_2} \ldots W_{i_n} | C_j) \cdot q_j}{\Sigma_{k \in L} P(W_{i_1} W_{i_2} \ldots W_{i_n} | C_k) \cdot q_k} \quad , \tag{3-4}$$

where $\{W_{i_1}, W_{i_2}, \ldots, W_{i_n}\}$ represents the sequence of keywords read from the document thus far, and q_j and L are defined in equation (3-3). Using the assumption of keyword independence in terms of context, proximity, and order, the probability in equation (3-4) is evaluated as

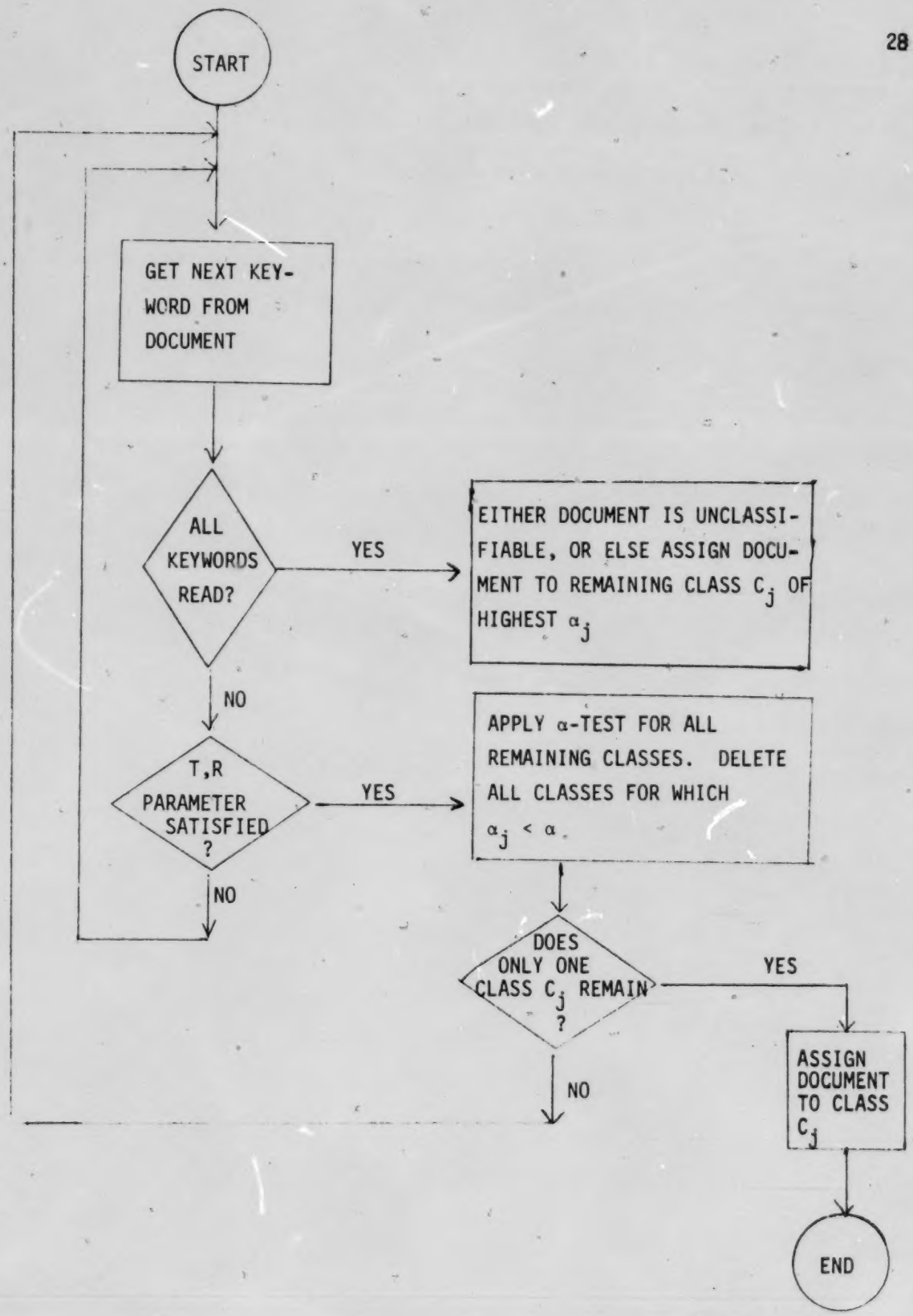

START
GET NEXT KEY-WORD FROM DOCUMENT
ALL KEYWORDS READ?
YES
EITHER DOCUMENT IS UNCLASSIFIABLE, OR ELSE ASSIGN DOCUMENT TO REMAINING CLASS C_j OF HIGHEST α_j
NO
T,R PARAMETER SATISFIED ?
YES
APPLY α-TEST FOR ALL REMAINING CLASSES. DELETE ALL CLASSES FOR WHICH $\alpha_j < \alpha$
NO
DOES ONLY ONE CLASS C_j REMAIN ?
YES
NO
ASSIGN DOCUMENT TO CLASS C_j
END

FIGURE 3.1 Flowchart of the Sequential Classification Method

$$P(W_{i_1} W_{i_2} \ldots W_{i_n} | C_j) = \prod_{k=1}^{n} P(W_{i_k} | C_j). \qquad (3\text{-}5)$$

Notice that the entire product in equation (3-5) need not be recalculated with each α-test, but only those factors need be multiplied corresponding to keywords read since the last α-test was made. The calculations in (3-4) and (3-5) may produce numbers which are very small (i.e., lead to computer underflow). This problem was circumvented by scaling each factor, since such scaling factors will cancel out in equation (3-5). The appropriate scaling factor depends upon the relative values of keyword frequencies and the maximum number of keywords expected to be read in any document, but for the experiments here a scaling factor of 150 was found to be satisfactory to prevent underflow.

Another problem which must be addressed in equations (3-4) and (3-5) is that any keyword may not occur at all in any of the sample documents of one or more class, and thus the corresponding frequencies in equation (3-1) are zero. Indeed this is to be expected, since a keyword should be associated with one class, or several classes, but it would be undesirable that a keyword be associated with high frequency with all classes, as this sort of keyword would be of dubious use for classification. A zero frequency or probability used in a pure form of Bayes law for equation (3-4) would then preclude any such class from being selected if such a keyword were detected. This is unrealistic for this task, however, since a keyword may occur near the beginning of the document which is either an incorrect indicator of the content of the document, or else indicates a different aspect of that document, i.e., another class. Instead of using the zero probabilities which arise in equations (3-1) and (3-2), a <u>default probability</u> is defined for all such cases to be substantially smaller

than the smallest nonzero probability found by equation (3-2) but large enough to allow a class to stay in contention in the α-test to be described below. Experiments which investigate the variation of the default probability for the SPIN documents will be discussed in Chapter 4.

Returning to the α-test indicated in Figure 3-1, this consists of testing whether

$$\alpha_j < \alpha, \text{ for } j = 1, 2, \ldots, L \qquad (3\text{-}6)$$

for the L classes still remaining in contention, where α is an input parameter to the sequential classification algorithm. If the α-test (3-6) applies to any of the L remaining classes, then this means that the sequence of keywords, $\{W_{i_1} W_{i_2} \ldots W_{i_n}\}$ read from the document so far do not sufficiently indicate C_j is a correct class, but do suggest other classes are appropriate for these keywords and for this document. Thus, as indicated in Figure 3.1, if test (3-6) applies for class C_j, then class C_j should be dropped. If one or more classes are dropped from subsequent consideration, then as previously explained, the class a priori probabilities in equation (3-3) are recalculated.

The parameter α is related to the probability that a document is misclassified. The a posteriori probability of class C_j given a sequence of keywords read from a document is defined to be α_j in equation (3-4). α_j is then bounded from below by parameter α, for otherwise the sequential test would have removed C_j from consideration. Experiments in Chapter 4 will show the situation to be quite complex, and the action of α is also dependent upon values of T and R, but generally if α is decreased the accuracy tends to improve since more keywords are read before classification is attempted. However, α cannot be decreased arbitrarily because as α becomes lower, more classes remain in contention and the entire document will have to be read before a classification decision is made.

Experiments in Chapter 4 will show how this trade off of selecting an appropriate value of α can be approached.

The process of reading keywords, applying the α-test, and dropping classes terminates in one of two ways.

1) The document is read entirely, and a number of classes remain as viable possibilities. If the number of these classes is sufficiently small, they may all be deemed applicable, or a subset with the largest α_j could be selected, but if there are too many classes remaining in contention, the document should be declared unclassifiable or should be placed in a general class.
2) All the classes have been eliminated by the α-test except one, and the document can be unambiguously assigned to that class.

As will be shown in subsequent chapters, this method is reasonably successful for assigning documents to a single class, but is not a very systematic way to approach multiple classification. Chapter 9 will present a modification of the sequential method which is much more appropriate for this more difficult problem.

3.6 Evaluation Criteria for Experiments

In Chapter 4 through 6, experiments will be described which evaluate the classification accuracy of the sequential classification algorithm presented here. The processing time and storage required for various document data bases is discussed, but they are generally determined by the size and description of the data base considered. The criterion evaluated is classification accuracy as the sequential algorithm operates under different conditions and uses various parameter values.

One problem in carrying out this evaluation is for documents which contain no keywords, or in general, contain fewer than T keywords. In this case, no α-test can be performed, and it is only fair to exclude such documents from any classification accuracy statistics. Of course, there should be few such documents, or this would indicate an inadequacy of the keywords to appropriately describe documents within the data base.

The number of correctly classified documents, the number of incorrectly classified documents, and the relative number of documents classified correctly are all specified when reporting experimental results. In the case of documents which are preclassified into multiple classes, (recall that the sequential algorithm forces only one class to be chosen), correct classification is recorded when that class appears among the multiple classes. If the document has been read entirely with no single class selected, then the class with the highest α_j at that point is arbitrarily selected for evaluation purposes. The problem of evaluation becomes very complex, however, when the classification algorithm selects multiple classes and these overlap but are not identical to the given preclassification multiple classes.

CHAPTER 4

EXPERIMENTS WITH THE SEQUENTIAL ALGORITHM USING THE SPIN DATA BASES

The SPIN data base is a collection of information notices or abstracts which are provided by the American Institute of Physics (AIP). Each abstract consists of a title, location, classification code, keywords or phrases, and bibliographic information. Each abstract is classified into one or more categories by means of a six character hierarchical classification scheme [35]. As many six-letter codes as deemed appropriate are assigned to each abstract by an AIP indexer. The first character of this code is any of seven different letters, which represent seven different classes. Extensive experimentation was done using these "root" classes (see Table 4.1). More than fifty classes exist at the next level, and hundreds of classes exist at the lowest level of the hierarchy. This extensive classification scheme which is provided as part of the SPIN data base offers a great potential for classification testing.

A subset consisting of 674 abstracts was selected from the December 1971 tape for initial experimentation. Preliminary processing of the abstracts was performed in order to facilitate subsequent processing. This preprocessing involved the deletion of non-essential header fields, mathematical and chemical notations, and special punctuation constructions.

The following sections discuss the sequential classification experiments which were conducted using the SPIN data base. The algorithm parameters and the experiment evaluation criteria used were discussed in Chapter 3. Section 4.1 examines the effect of the various input and default parameters on classification accuracy. Section 4.2 presents our experimentation with the sequential algorithm itself, and in Section 4.3, the keyword experiments are described.

CLASSIFICATION CODE	DESCRIPTION
B	General
D	High energy and nuclear physics
M	Atoms, molecules, and chemical physics
P	Fluids and plasmas
S	Solid state physics
T	Acoustics, optics, and cross-disciplinary physics
X	Astronomy and astrophysics

Table 4.1 SPIN Root Classes

4.1 Classification Experiments

The SPIN data base was the first to be classified using the sequential technique. Consequently, a major evaluation effort was undertaken to determine the effects of the input parameters and algorithm default values, and to ascertain the most appropriate values for these variables. For these experiments, five-hundred documents were selected and a set of 634 keywords was used. The methods for selecting and evaluating this keyword set are discussed in Section 4.3.

4.1.1 Default Probabilities

A default probability, as explained in Chapter 3, is simply a means to prevent premature elimination of categories. It involves the assignment of a positive but very small value as the probability that a keyword belongs to a class, when in fact, the keyword is not associated with the given class. Experimentation was needed to determine how sensitive the classification results were to different default probabilities, and what the optimum value for such a default value might be.

The initial investigation centered on the effect of various default values on the accuracy of the sequential algorithm if classification were postponed until the entire document was processed. The document was merely assigned to the class with the highest α_j level. Table 4.2 shows that larger default values give lower classification accuracy; a default value smaller than 5×10^{-5} caused underflow problems as such small probabilities are multiplied together.

The effects of the default value on accuracy were then studied using the sequential technique which does not require the entire document to be read. A 76% classification accuracy was achieved using the default value of 5×10^{-5}, and this accuracy was essentially unaffected as the default parameter was

DEFAULT VALUE	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY (EXCLUDED 12 DOCUMENTS WITH NO KEYWORDS)	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
1.5×10^{-3}	400	82.0	500
1.0×10^{-3}	402	82.4	500
5.0×10^{-4}	407	83.4	500
2.5×10^{-4}	409	83.8	500
5.0×10^{-5}	412	84.4	500

Table 4.2 SPIN Default Value Experiments on Entire Documents

varied about this value. Thus, in the sequel a default value of 5×10^{-5} was used consistently, although this figure may have to be adjusted for another data base.

4.1.2 α Values

A number of experiments were made in order to determine the best value of α, given the default value of 5×10^{-5}. For these experiments, the parameters T and R were held constant each at unity, and the value of α ranged from .07 to .30. Table 4.3 shows that the best α value was determined to be .13 with a classification accuracy of 77.7%. The maximum difference in accuracy did not exceed 2%. Thus, wide ranging α values did not have a significant effect on classification accuracy. Note, however, that the number of documents read entirely is dependent upon α, and in general decreases with increasing α as expected. The method used for calculating the class a posteriori probabilities α_j also underwent experimentation, and this is reported in Section 4.2.1.

It should be observed at this point that the performance of the sequential algorithm can be assessed by comparing Tables 4.2 and 4.3. When the document is read entirely, and the cumulative effect of all the keywords in the document are taken into account, in some sense this might be visualized as an upper bound on the performance of a sequential method when classification decisions are made in a similar way. Note that the classification accuracies of Table 4.3 are not significantly less than those of Table 4.2, thus indicating the effectiveness of the sequential method when only 17% of each document was read on the average. In the experiments reported in this chapter, the sequential classification accuracy can only be marginally improved if all 488 SPIN documents are classified. Since the performance of the sequential method is already close to this upper bound, it may be that these parameter changes or other modifications might yield

α	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY (500 DOCUMENTS, INCLUDING 12 DOCUMENTS WITH NO KEYWORDS)	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
.070	372	76.2	43
.100	376	77.1	39
.115	375	76.8	37
.130	379	77.7	33
.145	372	76.2	35
.200	370	75.8	35
.300	370	75.8	35

Table 4.3 SPIN Experiment with Different α Values, T=R=1

greater improvement in classification accuracy for other situations or data bases. Additional insight in this respect is provided in Chapters 5 and 6 as the sequential algorithm is applied to other data bases.

4.1.3 T and R Value Experiments

Modifying the values of the T and R parameters from a unity value did increase the accuracy of classification; but due to the larger number of documents which failed to contain enough keywords to pass the threshold, there was an increase in the number of documents which had to be read entirely. Table 4.4 illustrates a series of tests for parameters T = 2 and R = 1. Comparing this to Table 4.3 shows a slight increase in classification accuracy but with an accompanying increase in the number of documents read entirely. The percentage accuracies were modified upward since 34 documents with fewer than two keywords were excluded in this calculation, but note that this figure is included in the number of documents read entirely.

A number of experiments were conducted in which α was assigned a value of 0.13 as suggested by Table 4.3, and parameters T and R were varied, producing a wider range of accuracy fluctuations than did the α-value experiments. Table 4.5 shows some of the results of these experiments. Here the parameter T is more effective at increasing accuracy than is the R parameter. In these experiments reported in Table 4.5, the classification accuracy achieved was quite high, but at a cost of more documents read entirely. It might be pointed out, however, that these documents probably have too few keywords to be reliably classified by any method.

There appear to be two distinct future uses of the T parameter for arbitrary data bases. The first is to set the value appropriately high in order to achieve an improved classification accuracy. The second use is to identify some of those documents which must be classified manually because they contain an insufficient

α	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY (500 DOCUMENTS, INCLUDING 34 DOCUMENTS WITH FEWER THAN TWO KEYWORDS)	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
.13	371	80.0	58
.20	373	80.4	55
.25	374	80.6	52
.30	374	80.6	49
.35	376	81.0	44
.40	373	80.4	43
.45	374	80.6	42

Table 4.4 SPIN Experiments with α Values Where T=2, R=1

T	R	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY	NUMBER OF DOCUMENTS WHICH SATISFY THE T THRESHOLD	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
2	1	373	466	80.0	55
2	2	375	466	80.5	70
4	1	364	417	87.3	90
4	2	358	417	85.6	99

Table 4.5 SPIN Experiments Varying T and R with $\alpha = 0.13$

number of keywords. Of course, documents may be deemed unclassifiable for other reasons, e.g., documents in which the subject matter changes too frequently for even several classes to capture the objective of the document; a document must be at least somewhat homogeneous in order for any automatic classification method to be successful. By use of an increased T parameter, high accuracy could be achieved with those documents classified automatically, and also high accuracy could also be obtained on those classified manually. Moreover, when automatic classification of a particular document fails, the reason for this failure could be associated with that document, thus allowing manual classification to be done more easily. It should be emphasized that the percentage of documents to be classified manually should not become too large, i.e., it should not exceed 10-15% of the entire document set.

4.2 Experimentation With the Sequential Algorithm

During the course of this research, a number of modifications of the sequential algorithm described in Chapter 3 were investigated. These modifications resulted in only marginally increased classification accuracy, possibly because this accuracy was already close to the bound provided by reading the documents entirely. These modifications are documented here since they might prove more effective for some other situation or data base.

4.2.1 The α_j Calculation

The preliminary experiments on the SPIN data base not reported here, the a priori probabilities q_j of each class C_j were assumed equal. Referring back to Chapter 3, this corresponds to the assumption of equal q_j in equation (3-3) for subsequent use in Bayes rule. This corresponds to equation (3-4) for the a posteriori probability α_j for each class C_j given the sequence of keywords read from the document to be classified. When the a priori probabilities q_j were calculated using equation (3-3), the resulting α_j from equation (3-4)

became a true Bayes a posteriori probability. This improved the classification accuracy by only about 1% for the SPIN data base, but was used on all experiments described in this report.

4.2.2 The α Ratio Test

During the course of this research, it was observed that the average α_j value was higher when correct classification was made than when an incorrect identification resulted. This might be expected, since α_j should represent an approximate measure of confidence that the document has been correctly classified. Another observation involved the ratio of the two highest α_j values for classes still under consideration as the sequential algorithm was applied. A test was temporarily incorporated into the algorithm which compared the ratio between α_1 and α_2, where

α_1 = highest α_j value, and

α_2 = second highest α_j value, over those classes still under consideration.

A ratio threshold test of α_1/α_2 was used to determine if this additional information could lead to more accurate classification. If the ratio of α_1/α_2 did not exceed a prespecified threshold value which ranged from 5 to 100 then more keywords were read even though the usual α-test described in Chapter 3 was satisfied.

Many experiments were conducted with various combinations of α and α_1/α_2 ratio values. Although improvement in classification accuracy in the 76-79% range was obtained, the ratio test was eliminated from subsequent versions of the sequential classification algorithm. A study of the reasons why the α_1/α_2 ratio influenced correct classification lead to the useful concept of "noisy" keywords discussed in Chapter 8. The modified implementation described and tested in Chapter 9 may be considered a generalization of the ratio test.

4.2.3 The General Class

The SPIN general class B presents special problems for any automatic classification method. These documents do not belong to any well-defined class and the subject matter of these documents can range over widely diverse topics. The determination of a set of keywords for these general documents is a difficult problem. For example, keywords were selected to yield a reasonable indication of class B documents, but then these generally occurring keywords caused continuing difficulty as documents from other classes were misclassified.

Another approach to the automatic determination that a document belongs to a general class is to arrive at this decision when no suitable class can be found, especially if the entire document must be read. With this approach, no general class keywords need be selected, which may subsequently mislead the classification of otherwise classifiable documents.

A preliminary study of this type was conducted and is reported in Table 4.8. As it involves a keyword selection technique, this aspect of the study will be described in more detail in Section 4.3.3. Only six classes were identified for SPIN (with B deleted) and the sequential classification algorithm was applied as usual with a reduced set of keywords (initially 602) using parameters α = 0.085 and R=T=1. Only if the entire document were read was the classification algorithm modified. This modification is shown in Figure 4.1. If more than two classes remain in contention, the document cannot be specifically classified, and so it is placed in the general class. The ratio threshold is then applied, and if the keywords have indicated class C_1 sufficiently strongly, this classification is made. If not, then it was realized that some classes are associated more than others, and this must be taken into account at this point. A correlation study of the classes of the type described in Chapter 2 revealed that classes D and M were strongly correlated, and that classes M, P, and S were all strongly correlated, all in terms of indicative keywords. Based upon

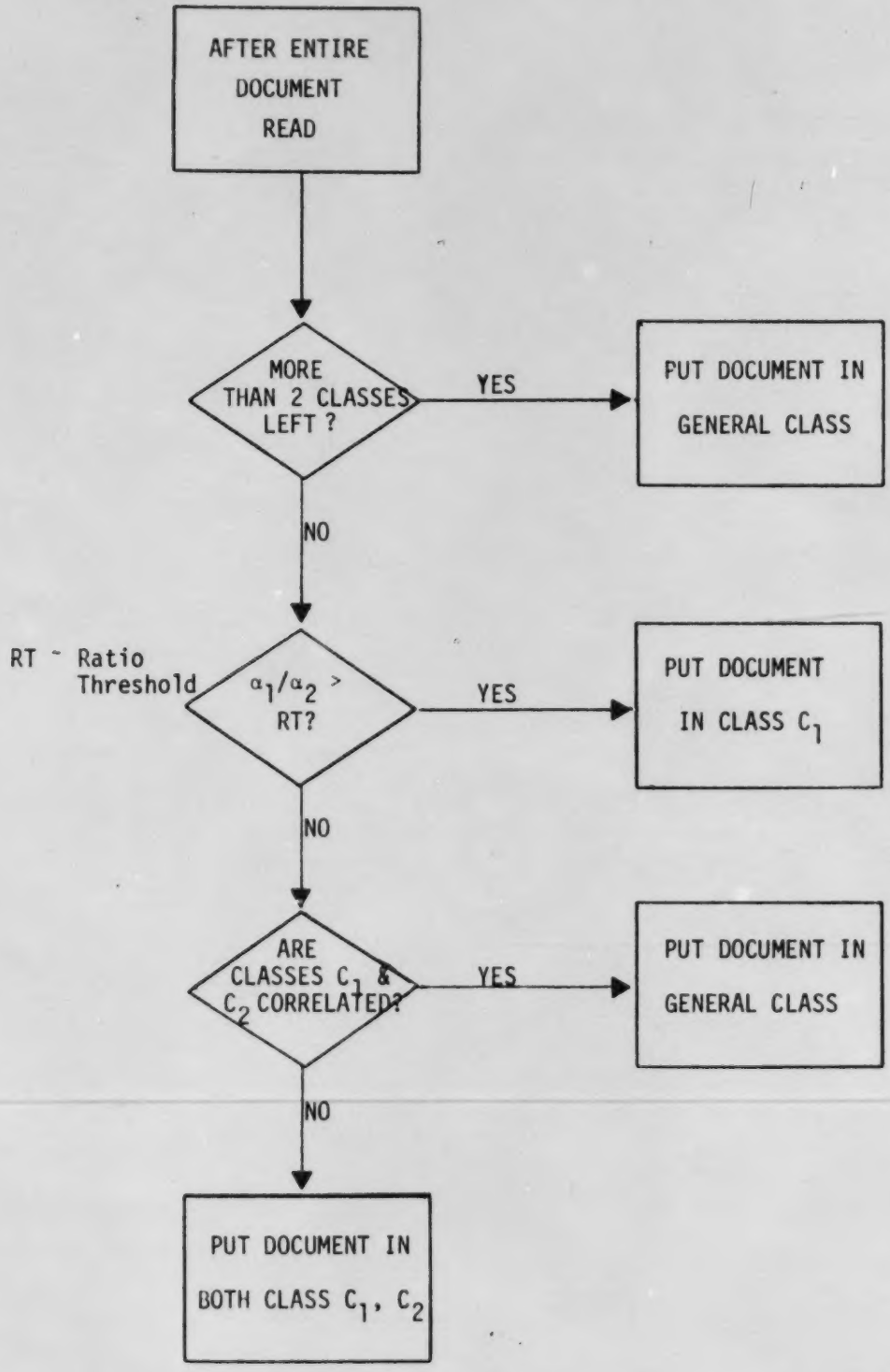


Figure 4.1 General Class Decision Procedure

this correlation information, the document is then either placed in the general class, or associated with both class C_1 and C_2.

This study produced the best classification accuracy for the SPIN data base when T=1 as shown in Table 4.8. Multiple class assignment has been achieved, as well as placement of the document within a general class. This was only a preliminary study, and again a more sophisticated approach to these problems of multiple class and general class assignment is given in Chapter 9.

4.3 Keyword Set Experiments

The manual and automatic keyword selection techniques described in Chapter 2 were applied in a number of ways to the SPIN document data. It will be shown how the "good-bad" keyword criterion iteratively led to improved keyword sets when used in conjunction with the sequential algorithm. These techniques would hopefully in the future be replaced by the more sophisticated "noisy keywords" methods developed in Chapter 8 and 9. Totally automatic keyword selection criteria without manual intervention or "good-bad" keyword feed-back will be shown to give surprisingly good results. These experiments were applied to the seven SPIN class data, but a similar study was also made with eleven class data in order to determine appropriate keywords. A clustering algorithm and evaluation criterion for comparing classifications from Chapter 2 were applied to the SPIN data. Since keywords are used as attributes in the clustering algorithm, the consistency of the results or stability of the clustering algorithm gives some indication of the appropriateness of these keywords. The clusters obtained can be matched against the classes originally chosen to check their validity or help in the selection of such classes. An example of the use of the evaluation criterion for comparing classifications will be given.

4.3.1 Initial Keyword Selection

Keywords for the SPIN documents with seven classes were initially selected based upon at least three occurrences within the set of 20 sample documents uniquely classified in each of the seven major SPIN categories, and non-occurrences in the stoplist. These keywords were found to be inadequate for sequential classification. Thus a return to the sample documents to obtain better keywords was made, and in order to enrich the abstracts with potentially important keywords, abstract titles were combined with the body of these documents.

A program was written which listed all of the non-stoplist words found in these 140 sample documents and exhaustively counted their occurrences in terms of relative frequencies in individual documents, overall frequencies in each class, and total frequency for the sample set. For each of the seven SPIN categories a list of words was selected; each word occurs in at least two documents and at least three times in all the documents at that class in the sample. In addition, some words of low frequency were chosen manually if they seemed indicative of the class.

The "good-bad" keyword criterion described in Chapter 2 was then iteratively applied to this keyword set obtained above. It was discovered that many words which were intuitively "good" actually degraded performance over 30 times within the 500 document run. Also not quite 10% of these original keywords substantially aided performance. This procedure resulted in a new set of 813 keywords which provided a respectable classification accuracy.

The set of 813 keywords was used with the sequential classification algorithm and the "good-bad" keyword criterion was applied iteratively. Those keywords which resulted in net degraded performance were eliminated, and the algorithm was reapplied to the 500 documents with the revised keyword list. This was

done several times until a total of 209 of the words were removed. Table 4.6 shows the results of these experiments.

Various values of T and R were utilized in this experiment with $\alpha = 0.30$ to get maximum benefit of each keyword set. As more keywords were removed using the iterative method, a point of diminishing return is observed as the number of keywords decreases beyond a certain point. The number of documents incorrectly classified decreases steadily, but a price is paid in terms of the number of documents which must be read in their entirety. As a result of this study, a final set of 634 keywords (related to the 636 in Table 4.6) was selected and used in most of the studies reported in this chapter. Prior to the selection of this keyword set, only about 300 documents of the 500 had been correctly classified.

4.3.2 Automatic Keyword Selection

The general guidelines for automatic keyword selection from Chapter 2 were utilized to apply to the seven class 500 SPIN documents. Although the results were somewhat inferior to that of the keyword set selected in Section 4.3.1, the classification accuracy is quite high considering that there was no manual intervention in the keyword selection as there was in that of Section 4.3.1.

The following specific selection criterion was applied to each distinct word in the 140 sample documents not on the stoplist.

1) Only consider words which occur at least three times within the sample documents, but occur in no more than three classes of sample documents.
2) If the word occurs only within the 20 documents of one class, accept that word.
3) If the word occurs in exactly two classes of sample documents, at least 60% of the occurrences should be in one class of documents.

TEST DESCRIPTION	NUMBER OF DOCUMENTS CORRECTLY CLASSIFIED	NUMBER OF DOCUMENTS INCORRECTLY CLASSIFIED	ENTIRE DOCUMENT READ
T=2, R=2 813 Keywords	336	140	24
T=6, R=2 813 Keywords	338	126	36
T=4, R=2 712 Keywords*	331	62	107
T=1, R=2 712 Keywords*	351	95	54
T=1, R=2 636 Keywords*	362	62	76
T=1, R=2 604 Keywords*	357	44	99

*Keywords Deleted on Basis of Iterative Criterion

Table 4.6 Sequential Classification Using Iteratively Reduced Keywords for SPIN Documents ($\alpha = 0.30$)

4) If the word occurs in exactly three classes of sample documents, at least 50% of the occurrences should be in one class of documents, and no more than 13% of the occurrences should be another class of documents.

Table 4.7 shows the effect of this automatic keyword selection criterion, together with several iterations of the "good-bad" keyword criterion discussed in Section 4.3.1. These results compare quite favorably to those given in Table 4.6, and provide encouragement to the concept of automatic keyword extraction.

4.3.3 General Class Experiments

Section 4.2.3 showed an approach to deal with both the problems of a general class and assignment of multiple classes. Experiments were conducted using this approach together with the use of the iterative "good-bad" criterion, and the results are presented in Table 4.8. The set of 602 keywords initially identified in Table 4.8 was obtained from the 634 keywords by deleting those keywords associated with the general class.

4.3.4 Experiments on Eleven Classes of SPIN

In order to contrast to the seven classes of SPIN studied thus far, it was desired to identify a larger number of classes and generate keywords for these classes. The next section, 4.3.5, describes the clustering techniques and experiments by which SPIN classes were identified, and eleven classes resulted as a coherent well-separated group from this study. As in Section 4.3.2, all the keyword sets for these eleven classes were generated automatically.

Eight different sets of keywords were generated, and the details of these keyword selection techniques are given in Appendix A, but discussed generally in Chapter 2. The keyword sets are labelled A through H, and satisfy subset

TEST DESCRIPTION	NUMBER OF DOCUMENTS CORRECTLY CLASSIFIED	NUMBER OF DOCUMENTS INCORRECTLY CLASSIFIED	NUMBER OF DOCUMENTS READ ENTIRELY
520 Keywords	324	34	142
444 Keywords*	342	80	78
420 Keywords*	340	93	67

*Keywords deleted on basis of iterative "good-bad" criterion.

Table 4.7 Automatic Keyword Selection, for SPIN Documents, $\alpha = 0.30$, T=R=1.

NUMBER OF KEYWORDS	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY	NUMBER OF DOCUMENTS WITH ONE OR MORE KEYWORDS	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
602	379	485	78.1	40
599*	379	485	78.1	41
596*	382	483	79.1	43
592*	382	481	79.4	46
589*	379	480	79.0	50

*Keywords deleted on basis of iterative "good-bad" criterion.

Table 4.8 SPIN General Class Experiments with Modified Keyword Set, with $\alpha = 0.085$, RT = 1.5, T=R=1; 6 classes.

relations:

$B \subset A \subset C$, $E \subset D$, and $H \subset G \subset F$.

The results of classification experiments on these keyword sets are given in Table 4.9. Thes best classification accuracy obtained was 72% for keyword set E, and 71% for set H with fewer documents read entirely.

4.3.5 SPIN Cluster Generation Study

The sequential classification algorithm initially requires a classification be given together with a set of keywords, and a sample set of documents with their respective classes. In the case of the SPIN data base, tests of the algorithm have involved comparing the classification results with the classification included in the SPIN base. However, since such a manually provided initial classification may not always be available for every application, there is a need for an automatic classification generating algorithm to apply to a set of sample documents.

Chapter 2 has discussed a modified version of Rocchio's algorithm [29], which has the advantages of simplicity and flexibility, as a number of parameters are provided to allow it to adapt to different clustering problems. This modified algorithm was applied to the SPIN data base in order to identify interesting classes.

Three series of experiments were conducted. The first two utilized the original set of 565 keywords as attributes in the clustering algorithm. The objects to be clustered were a collection of 96 documents selected at random from the base of 500, and the 140 sample documents, from which the 565 keywords have been chosen. The third series of experiments utilized the 140 sample documents together with the 634 keywords mentioned in Section 4.3.1.

Table 4.10 shows the results of these three series of experiments. In each experiment, parameters are changed, and natural clusters of documents

KEYWORD SET*	NUMBER OF KEYWORDS	α	NUMBER CORRECT (of 500 possible)	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY	NUMBER OF DOCUMENTS WITH NO KEYWORDS
A	481	.13	317	64.2	28	6
B	437	.13	330	67.9	51	14
C	494	.13	326	65.7	24	4
D	534	.13	320	64.0	8	0
		.10	322	64.4	23	0
E	433	.13	349	72.0	48	15
		.10	346	71.3	63	15
F	661	.13	304	60.8	1	0
		.10	314	62.8	8	0
G	620	.13	346	69.3	6	1
		.10	354	70.9	16	1
H	571	.07	355	71.1	42	1

*Keyword set relationships

$B \subset A \subset C$

$E \subset D$

$H \subset G \subset F$

Table 4.9 Experiments on Eleven Classes of SPIN Using Automatically Selected Keyword Sets, R=T=1.

are identified, where the number of clusters is not determined a priori. The classification K_1 which results is compared to K_2, the given seven classes for the SPIN abstracts. In Chapter 2, a method for comparing two classifications was presented, resulting in $||K_1 - K_2||/T$, which is a measure of the normalized distance between two classifications K_1 and K_2.

The results indicate that six clusters provide a better agreement with the seven classes which are currently being used. This can be partly attributed to the fact that the SPIN class "B" has only 23 documents in the collection of 500 documents, so that the manually provided classification in reality consists of only six major categories. This is accentuated by the fact that "B" is a general "catch-all" category.

Note that the two classifications are closer using the 140 document set than with the 96 random documents. This can be explained in that it was these documents which provided both the 565 and 634 keyword sets used for cluster generation, and not all the keywords were even found in the 96 randomly selected documents. Also, these documents were all uniquely classified by the seven SPIN categories, while many of the 96 random documents were assigned multiple SPIN classification.

In the second series of experiments, the natural clustering of the eleven classes studied in Section 4.3.4 is apparent from Table 4.10. The third series of experiments, although incomplete, show the classifications to be in much closer agreement than those obtained by any of the other experiments, thus illustrating again the superiority of the 634 keyword set over the original 565 keyword set.

4.3.5 Keyword Stability

In all the experiments on SPIN described in this chapter, the keyword frequencies were obtained over the sample set of documents. The question

SERIES	EXPERIMENT NUMBER	NUMBER OF CLUSTERS	$\|\|K_1-K_2\|\|/T^*$
96 Documents-Random 565 Keywords	1	4	.71
	2	6	.66
	3	7	.73
140 Documents-SPIN Sample Set 565 Keywords	1	5	.59
	2	10	.61
	3	11	.59
140 Documents SPIN Sample Set 634 Keywords	1	7	.50
	2	8	.53

*$||K_1-K_2||/T$ measures the normalized distance between two classifications K_1, K_2. K_2 is the given classification for the SPIN abstracts and consists of 7 classes.

Table 4.10 Comparison of Classifications from Clustering Algorithm to SPIN Categories

α	NUMBER OF DOCUMENTS CLASSIFIED CORRECTLY	% CORRECT	NUMBER OF DOCUMENTS READ ENTIRELY
.13	364	74.6	31

Table 4.11 SPIN Experiments with Keyword Frequencies Counted Over All 500 Documents, α = 0.13, T=R=1.

naturally arises as to the stability of this frequency count, and if effectively different counts would be obtained if the entire 500 document set were utilized.

An experiment was conducted in which frequency counts were produced in this way before classification was made. Table 4.11 shows the results of this experiment, and these results should be compared to those found in Table 4.3. The keyword set, 634 words, was the same as for Table 4.3, thus the same twelve documents of 500 contained no keywords at all. It is clear that the frequency counts are very stable, since the classification accuracy is essentially unchanged.

CHAPTER 5

COMPARISON OF THE SEQUENTIAL AND DISCRIMINANT ANALYSIS CLASSIFICATION TECHNIQUES

The automatic document classification research of Williams [42,43] using his discriminant analysis method has produced some of the highest classification accuracies to date. Thus, his computer programs and data base were acquired in order to obtain a comparison between his method and the sequential classification method. The two methods were compared using both the solid state abstracts used by Williams and the SPIN abstracts discussed in Chapter 4. The results of these experiments and some general comparisons are presented in this chapter.

5.1 Discriminant Analysis Approach

In the discriminant analysis technique, each category is represented by a vector of mean keyword frequencies. This vector is derived from sample documents manually assigned to that category. A new document is classified by comparing the frequencies of the keywords in the document to the mean vector of each category. The document is then put in the class whose mean vector is closest to the keyword frequency vector for the document. The vector comparison requires a set of weighted coefficients which are computed by a formula which requires the use of N x N matrices, N being the number of keywords. In addition, this transformation requires a matrix inversion of the N x N matrix. Due to the time and storage difficulties involved in such matrix manipulations, only a relatively small number of keywords can be used for document classification. Accordingly, Williams chose 48 keywords to describe a 420 document sample of the solid state abstracts. These 48 keywords were used as the first set of keywords for the comparison experiments.

5.2 Solid State Experiments

The solid state collection consisted of 1743 abstracts, but classifications were only available for 420 of the documents of this collection. These 420 documents, with known classifications in three categories, constituted the data base for the solid state experiments. Two different experiments were performed with each of three different keyword sets. In the first experiment for each keyword set, the frequency counts and distributions necessary for each method were computed using all 420 documents. In the second experiment, only 120 documents, 40 from each class, were randomly chosen to obtain the necessary frequency counts and distributions. Thus, the remaining 300 documents could be used as a set of test documents.

5.2.1 The 48 Keyword Set

The 48 keyword set was devised by Williams. Using these 48 broad and very frequent keywords, Table 5.1 shows that discriminant analysis did well in the first experiment with 85% classification accuracy. When the 120 sample set was used, the results for discriminant analysis on this small sample of 120 documents, 48 keywords were not as impressive: 88% on the 120 sample documents, 67% on the 300 test documents, giving 75% overall. Williams had reported 88% on sample documents, 79% on test documents using 1743 documents from the solid state abstracts data base with a sample of 245 documents. Williams himself concluded that the number of documents per category in the sample is very important, and a large number is necessary if the documents are as short as the solid state abstracts, possessing an average of 90 words per document. Since the 420 document sample was carefully chosen, the degradation in performance due to a smaller sample was to be expected.

In contrast to the above results, classification based on the entire 420 document set for the sequential analysis method yielded a smaller classification accuracy as seen in Table 5.1. Yet in the second experiment the

KEYWORD SET	DISCRIMINANT ANALYSIS	SEQUENTIAL ANALYSIS $\alpha = .05$	SEQUENTIAL ANALYSIS ENTIRE DOCUMENT READ
48 Keywords			
Doc. No. = 420	85%	70%	74%
Sample = 120	88%	88%	90%
Test = 300	67%	79%	82%
Total = 420	75%	81%	85%
67 Keywords			
Doc. No. = 373	87%	88%	91%
Sample = 113	74%	93%	95%
Test = 260	52%	83%	85%
Total = 373	59%	86%	88%
75 Keywords			
Doc. No. = 404	87%	80%	84%
Sample = 119	82%	94%	93%
Test = 285	57%	84%	86%
Total = 404	64%	87%	88%

Table 5.1 Comparison of Sequential and Discriminant Analysis Methods for Solid State Abstracts

performance of the sequential method was actually superior to that of Williams, again emphasizing the point that a small sample set of documents can be utilized more effectively with the sequential approach. It can be further observed that reading the entire document here does not increase the classification accuracy appreciably.

A study of the 48 keyword set revealed that six general and very frequent keywords occurred at the beginning of the documents, and thus mislead the sequential method. The "good-bad" keyword analysis criterion showed that although the six bad keywords occurred frequently over all three categories, they tended to favor one and eliminate the other categories. The effects of these six keywords were lessened in the 120 document sample as they were not as frequent or as weighted for one category. In order to correct the inequity caused by these six keywords, two other sets of keywords were devised and tested.

5.2.2 The 67 Keyword Set

Since the broad and very frequent keywords led to misclassification, a new set of 67 keywords was selected according to the following criteria: only those keywords which favored one class at least 60% of the time were chosen. Only a portion of the 48 original keywords met this criterion. The number of tokens per category for the larger sample dropped from 1200 for the 48 keywords to about half this figure for the 67 keywords. Due to the more specific and less frequent keywords, only 373 of the 420 documents contained keywords from the 67 set.

Discriminant analysis (Table 5.1 on 67 keywords) performed slightly better on this keyword set for the 373 document sample with 87% but yielded a disappointing 59% overall with the smaller sample. In contrast, sequential analysis achieved 88% and 86% respectively, and classification took place much faster than on the 48 keyword set.

5.2.3 The 75 Keyword Set

A compromise between the 48 and 67 keyword sets was necessary to increase the number of documents having keywords. A set of 75 keywords was formed by adding keywords from the 48 keyword set. The addition of these eight keywords allowed 404 documents to contain at least one keyword.

The addition of these eight keywords increased the overall performance of the discriminant analysis method by 5% in the second experiment, but did not affect the performance of sequential analysis. The first experiment showed no change for discriminant analysis, but sequential analysis performance decreased by 8%. Thus by adding eight broad keywords, two of which occurred frequently in more than one class, sequential analysis was adversely affected.

5.3 SPIN Experiments

Further comparison experiments were conducted using the SPIN data base, the data base most extensively used in the testing of the sequential method and reported in Chapter 4. Using six categories and a 67 keyword set, 434 documents were divided into a 118 document sample and a 316 document test set. Table 5.2 shows the sequential method to be superior to that of discriminant analysis, with special problems for the latter approach when the test documents are encountered. These results essentially confirm those found in the solid state experiments reported above.

5.4 Conclusions

The experiments on the solid state data base indicate that discriminant analysis can be an effective classification method for a small number of classes and keywords if the document sample is carefully chosen; the keyword set is not crucial once a handpicked sample has been selected. Sequential analysis, on the other hand, demands keywords which are particular to one class; the document sample may be randomly chosen.

	DISCRIMINANT ANALYSIS	SEQUENTIAL ANALYSIS
Sample = 118	87%	93%
Test = 316	44%	74%
Total = 434	55%	79%

Table 5.2 Comparison of Sequential and Discriminant Analysis Methods for SPIN Data Base

Given a data base, representative keywords will be easier to come by than representative documents. The former requires frequency counts on a small sample, any sample, of documents. The latter requires a knowledge of the subject matter of the content of all the documents in the data base. For an automatic system having a minimum of human tuning and effort, sequential analysis is superior because

1) no subjective choices of what constitutes a representative document are necessary, and
2) the criteria for what constitutes a good keyword can be programmed as discussed in Chapter 2.

Before classification can take place in discriminant analysis, a number of costly matrix manipulations are necessary to find the discriminant characteristics for each class. Although this should be a one-time calculation if the sample set of documents is a representative one, it may be necessary to find the best sample set thus incurring further costly runs. Depending on the characteristics of the matrix, a great deal of underflow and computer time may be involved in calculating the discriminant characteristics before classification can take place. Sequential analysis requires no similar set-up, and keywords or new sample documents can be added or changed at little cost.

Actual time comparisons depend, of course, on both the document sample and the keyword set. In our experiments, Williams' method took about 25% longer to classify the documents. In addition, sequential analysis does not require that the entire document be in machine-readable form. New documents can be classified after only a portion of the document is entered at a teletype.

Any matrix technique limits the number of keywords, classes, and sometimes documents which may describe a data base. In sequential analysis, the time and storage only increase linearly with the number of keywords and classes.

CHAPTER 6

SEQUENTIAL CLASSIFICATION EXPERIMENTS USING THE KUCERA-FRANCIS AND EPILEPSY DATA BASES

This chapter discusses the experiments that were performed on the Kucera-Francis and Epilepsy data bases. The Kucera-Francis data base is a large diverse collection of full text documents classified into 17 categories. The Epilepsy data base consists of abstracts for which a hierarchical classification scheme is provided. This data base was partitioned into both 9 and 27 classes for experimentation purposes. The experiments performed were similar: parameters T, R, and α were varied and the keyword sets were modified in order to improve classification accuracy.

PART I: The Kucera-Francis Experiments

The Kucera-Francis data base is a collection of English language text in computer-readable form originally assembled at Brown University in 1964 [25]. The corpus consists of a million words of natural language text, and is divided into 500 samples of approximately 2,000 words each. The documents were constructed by combining journalism articles so that this length constraint would be met. Although an initial classification scheme was available, the categories were not suitable for automatic document classification since these categories reflect the type and source of the document, rather than its subject content. The following sections discuss the methods used for manual classification, keyword selection, and the classification experiments.

6.1 Data Base Classification and Keyword Selection

A sample set of 100 documents was randomly selected and manually classified. Seventeen categories were identified during this process. The remaining 400

documents, which comprise the test set, were also manually classified. The categories and number of documents classified into each category are presented in Table 6.1. Many of the documents were classified into more than one of the categories. Non-overlapping categories could not be devised since the documents themselves are not homogeneous. That is, a document can abruptly change topic areas due to the artificial method of document creation described above. Only 59 of the sample documents were classified uniquely into one category, 25 documents were assigned to exactly two of the categories, with the rest of the documents assigned to three or more. One-hundred-forty-one documents were classified as fiction and thus were not a part of the document base for either the sample or test set. Twenty-three other documents were also eliminated because they were too diverse; these journalism articles completely changed subjects many times, and were classified into five or more classes. After eliminating these documents, the sample set consisted of 70 documents, and the test set contained 266.

Keywords were manually selected from the documents in the sample set on an intuitive basis. Derivatives of these keywords were added, as were other intuitively chosen keywords that were not in the sample set. In each case, the keyword frequency was checked to determine if the keyword occurred at all, and if it occurred frequently enough to be in documents other than the sample set. If these two criteria were met, the keyword was added to the keyword list. In all, 2,133 keywords were selected.

6.2 Sequential Classification Experiments

A series of experiments were conducted on the data base of Kucera-Francis documents. For the experiments that were conducted, two different algorithms and three different frequency tables were used. The algorithm which was used for the experiments in Table 6.2 is a modified form of equation (3-4) in which the q_k in the denominator is equal for all remaining classes.

CLASSIFICATION	NUMBER OF DOCUMENTS	
	Sample Set	Test Set
A: Religion	66	29
B: International Affairs	11	48
C: Law	4	17
D: Fine Arts	13	52
E: Marriage and Family Relations	3	16
F: Medicine	4	17
G: Education	7	28
H: History	3	37
I: Social Science	6	55
J: Biographies	6	36
K: Local Politics	10	39
L: Business and Finance	12	42
M: Technology and Popular Science	3	28
N: Physical Science	7	19
O: Mathematics	3	10
P: Sports	5	22
Q: Natural Science	6	22
X: Fiction	31	139

Table 6.1 Kucera-Francis Classification Scheme
Determined by Manual Classification

The first experiment (see Table 6.2) was the sequential classification of the sample set. Of the 70 documents in the sample, the sequential algorithm successfully classified 69. Since the keywords were primarily selected from the documents in the sample, it was anticipated that the classification accuracy for the sample set would be higher than for the test set. This was verified by the subsequent experiments.

The second experiment was the first attempt at classifying the test set of 266 documents. For this experiment α = .06 and T=R=1, and the classification accuracy was 44%. There is an average of 62 keywords per document, and for the first four experiments, none of the documents were read entirely.

For the third experiment, the number of documents to be classified was reduced to 258; this number then remained constant throughout the experiments. For this experiment, T = 4, which meant that classification was postponed until at least four keywords were processed. Classification accuracy was not improved over the second experiment, however. For the fourth experiment the α value was increased to .30. Analysis of the classification output revealed that, in this diverse data base, classification was attempted after too few keywords were processed. Consequently, an α value of .03 or .06 was used in the subsequent experiments. The frequency table for the experiments in Table 6.2 was derived from the sample set and was not modified until the fifth experiment. Due to the experience gained with the classification algorithm and based on the good-bad counts provided by the classification program, several keywords in the table had their frequency-counts set to zero. This was equivalent to deleting the keyword from the keyword set. The result was that the total number of keywords with zero entries in the table increased from 44% to 53% resulting in improved classification accuracy.

EXPERIMENT NUMBER	DOCUMENT SET, NUMBER OF DOCUMENTS	T	R	α	% CLASSIFIED CORRECTLY
1	Sample 70	1	1	.06	98.6
2	Test 266	1	1	.06	44.0
3	Test 258	4	1	.06	42.6
4	Test 258	1	1	.30	38.8

Table 6.2 Kucera-Francis Experiments Using Modified Sequential Algorithm and Frequency Table Derived from Sample

For the experiments indicated in Table 6.3, the algorithm (3-4) was used. Experiment 5 can be compared directly to experiment 2, with experiment 5 showing a drop in accuracy of about 6%. Subsequent experiments, however, more than regained this loss in accuracy.

In order to obtain more information about the nature of the data base, it was decided in experiment 6 to process all the keywords in each document before attempting classification. This was achieved by setting $\alpha = 0$; i.e., classes were not dropped as the keywords were processed. This resulted in only 24% accuracy, indicating the very heterogeneous nature of the documents, and the need for improvement in the keyword set.

For experiments 7 and 8, the T parameter was assigned the value two. This was done to increase the number of keywords processed before classification was attempted. The results range from 44% and 46%, indicating a definite improvement.

A final experiment was conducted in which the keyword frequency table was generated from the combined sample and the test sets of documents. This was done in order to determine the stability of the keyword frequencies. The resulting classification accuracy was 68.7% (see Table 6.4). This substantial improvement is accounted for by the fact that all the keywords in the frequency table were associated with at least one class. Thus, every keyword contributed to the classification of each document; whereas previously, approximately 50% of the keywords did not occur in the sample or were eliminated from the frequency table as being "noisy".

6.3 Summary of Kucera-Francis Experiments

The Kucera-Francis data collection is relatively keyword rich when compared to the SPIN data base. Consequently, a major effort was made to find techniques for capitalizing on this feature. Merely increasing the T parameter (experiment 3) proved to be too simplistic an approach. Also, as

EXPERIMENT NUMBER	DOCUMENT SET, NUMBER OF DOCUMENTS	NUMBER OF DOCUMENTS READ ENTIRELY	T	R	α	% CLASSIFIED CORRECTLY
5	Test, 258	0	1	1	.06	37.6
6	Test, 258	258	1	1	.00	23.6
7	Test, 258	2	2	1	.03	44.2
8	Test, 258	3	2	1	.03	46.5

Table 6.3 Kucera-Francis Experiments Using Unmodified Sequential Algorithm and Modified Frequency Table

EXPERIMENT NUMBER	DOCUMENT SET, NUMBER OF DOCUMENTS	# DOCUMENTS READ ENTIRELY	T	R	α	% CLASSIFIED CORRECTLY
9	Sample and Test, 328	1	2	1	.03	68.7

Table 6.4 Kucera-Francis Classification Experiment Using Frequency Table Based on Entire Document Collection

experiment 6 dramatically showed, processing the entire document resulted in the poorest classification accuracy of any of the experiments (24%). So, for this diverse data collection, an approach which called for the processing of 5 to 10 keywords per document (out of an average of 62) proved to be the best. Several keyword sets were used, subsequent sets generated by eliminating keywords indicated as "bad" by the sequential algorithm results. The elimination of these bad keywords resulted in improved accuracy. A new approach for the identification of poor keywords has been studied, and the results of this study will be presented in Chapters 8 and 9.

The manual classification of the entire data base led to the identification of three new classes which were not represented in the sample set. This is a potential problem with the sequential approach, since whenever a sample is used to determine the classification structure, some of the classes may not be well represented in the sample. These new classes are: entertainment, food and nutrition, and military. They are not listed in the classification scheme (Table 6.1) since they were not used in the experiments, and no keywords were derived for them.

PART II: Experiments with the Epilepsy Data Base

The Epilepsy section of the National Institute of Neurological Diseases provides tapes of Epilepsy Abstracts to qualified information centers. The Epilepsy Data Base used in conjunction with the sequential classification study corresponds to the tape of abstracts from 1969. This data base is published in hardcover by the National Institute of Neurological Diseases and Stroke as *Epilepsy Abstracts* Vol. 2, Nos. 1-12, Abstracts 1-1240. The classification study used 1233 abstracts, of which the average length was 250-300 words.

6.4 Overview of the Epilepsy Data Base

Among the characteristics of this data base, the following features make it particularly suitable for experimental classification.

1) It is pre-classified into 28 classes, each document having only one associated class. These classes are shown in Table 6.5. In the study, 27 of these classes were used; class 1.1 was discarded, as the documents were too general for keyword selection.
2) The data base is supplemented with a keyword set associated with each document. These keywords were determined by biomedical specialists and based on the information contained in the original papers from which the abstracts were derived. For this reason, the set of keywords associated with a document need not actually appear in the abstract.
3) A second level of classification is supplied: a 9-element generic class. These 9 classes each contain 1 to 4 subclasses comprising the total 28 classes. The entire hierarchical classification system and the number of documents associated with each class are listed in Table 6.5.

For the version of the data base that was used in the classification experiments, only 57% of the keywords are associated with one category, 17% are associated with two categories, and 87% correspond to four or fewer categories. This means that a keyword, on the average, is associated with 2.4 of the 27 classes. When only 9 classes are considered, a keyword is associated with 1.8 of the classes.

The results of this investigation suggested that the first attempt at automatic classification of this data base be done on all 27 classes using a proper subset of the keywords. This subset and the a priori probabilities needed were derived from a sample set.

		Number of Documents
1.0	General Aspects	7
	1.1 Books, Symposium, Congress, and Proceedings	
2.0	Basic Sciences	7
	2.1 Physiology	125
	2.2 Biochemistry	82
	2.3 Experimental Seizures	28
3.0	Seizures	7
	3.1 Clinical Neurophysiological Investigation	47
	3.2 Clinical Aspects	103
	3.3 Case Reports	91
4.0	Etiology	39
	4.1 Etiopathology	10
	4.2 Seizures Associated With Other Diseases	156
5.0	Genetics	18
6.0	Systemic Changes Related to Seizures	7
	6.1 Biochemistry and Metabolism	12
	6.2 Psychophysiology	5
7.0	Diagnostic Aids	6
	7.1 Electroencephalography	135
8.0	Psychological, Sociological, and Epidemiological Aspects	12
	8.1 Psychiatric and Psychological Aspects	44
	8.2 Social and Vocational Aspects	11
	8.3 Forensic Aspects	6
	8.4 Epidemiology	8
9.0	Treatment	13
	9.1 Experimental Pharmacology	56
	9.2 Medical Treatment	177
	9.3 Surgical Treatment	19

Table 6.5 Hierarchical Classification System for Epilepsy Data Base

6.5 Selection of the Sample and Keyword Sets

In order to apply the sequential classification algorithm, the a priori probabilities of keywords within each class were needed. A sample set was necessary to generate these probabilities. Since prior knowledge of the distribution of documents over the 27 classes was available, we decided to select the sample set to reflect this proportion; i.e., the number of sample documents of a given class were chosen so that the class contributed the same percentage to the sample as to the entire data base. The sample was chosen to represent approximately 20% of the data base. The 245 abstracts in the sample were also selectively chosen to guarantee that they would not generate too small a subset of the 1,751 keywords; this choice resulted in an 832 word subset.

6.6 Sequential Classification Experiments

The following sections describe the classification experiments performed using both the 27 classes and the 9 classes, and the results of these experiments.

6.6.1 Experiments Using 27 Classes

For the experiments with 27 classes, 500 documents were classified, the value of R was unity, and the number of keywords was 832. The value of α ranged from .05 to .13 and several iterations were made on this initial keyword set to eliminate noisy keywords.

In experiment 1, $\alpha = .13$ and $T = 1$; the resulting accuracy was 35.8% with 16 abstracts read entirely (see Table 6.6). At this point, the keyword set was reduced to eliminate misleading keywords. These keywords were selected from:

i) an intuitive choice of "bad" keywords, and

ii) the "good-bad" criterion previously described.

The first reduced keyword set contained 826 words and resulted in improved accuracy for experiment 2. Since the number of abstracts read entirely was 38 in experiment 2, the T parameter was set to 2 and α reduced to .10. A 48%

EXPERIMENT NUMBER	NUMBER OF DOCUMENTS READ ENTIRELY	T	R	α	NUMBER OF KEYWORDS	% CLASSIFIED CORRECTLY
1	16	1	1	.13	832	35.8
2	38	1	1	.13	826	43.3
3	74	2	1	.10	826	47.9
4	116	2	1	.05	826	46.4
5	76	2	1	.10	817	48.0
6	90	2	1	.10	794	45.2
7	68	2	1	.10	807	43.0

Table 6.6 Epilepsy Experiments Using 27 Classes and 500 Documents

accuracy resulted in experiment 3, while the number of documents read entirely was 74. A further decrease in α proved to be detrimental. In experiment 4, where α = .05, the accuracy was 46.4% with 114 abstracts (almost one-fourth of the total) read entirely. At this point, very little further improvement seemed possible given the same keyword set. Therefore, the keyword set was again revised by eliminating keywords. For experiment 5, an 817 keyword set was used, and α = .10; the resulting classification accuracy was 48%, essentially the same as experiment 3. Thus for experiment 6, 23 more keywords were eliminated, yet classification accuracy was only 45% with 90 documents read entirely. In the final experiment over 27 classes, a keyword set of 807 words was selected which did not improve classification accuracy but did reduce the number of documents read entirely to 68.

Two possibilities were suggested to explain the lack of further significant improvement:

1) the keywords were poor indicators of the classes, or

2) the keywords used cannot sufficiently distinguish so many classes.

Clearly, these possibilities are not mutually exclusive. We decided to take advantage of the hierarchical structure of the data base by applying the algorithm to 9 classes using the generic classification, and thus test this second explanation.

6.6.2 Experiments Using Nine Classes

In order for the sequential algorithm to classify the Epilepsy documents into nine classes, the a priori frequencies in the keyword frequency table had to be recalculated. The 807 word keyword set which had been used in experiment 7 was used for experiment 8. Table 6.7 shows the resulting classification accuracy to be 40% with 56 documents read entirely. Setting α = .05 for experiment 9 not only significantly increased the number of documents read entirely to 114, but also decreased the classification accuracy to 35%.

EXPERIMENT NUMBER	NUMBER OF DOCUMENTS READ ENTIRELY	T	R	α	NUMBER Of KEYWORDS	% CLASSIFIED CORRECTLY
8	56	2	1	.13	807	40.2
9	114	3	1	.05	807	35.2
10	87	2	1	.13	817	52.1
11	19	2	1	.30	817	54.5
12	23	2	1	.40	817	54.3

Table 6.7 Epilepsy Experiments Using Nine Classes and 500 Documents.

EXPERIMENT NUMBER	NUMBER OF DOCUMENTS READ ENTIRELY	T	R	α	NUMBER OF KEYWORDS	% CLASSIFIED CORRECTLY
13	79	2	1	.10	1751	44.9

Table 6.8 Epilepsy Experiments Using 27 Classes, 500 Documents, and 1751 Keywords.

At this point, a change in the keyword list (which had made only marginal differences in the 27 class experiments) to the one used in experiment 5 on 27 classes resulted in substantial improvement. In experiment 10, with α reset to .13 and using the 817 word keyword list, 52.1% accuracy was achieved with 87 documents read entirely. In experiment 11, α was increased resulting in 54.4% accuracy with only 19 documents read entirely. A final increase in α to .40 in experiment 12 showed no further improvement.

6.7 Summary of Epilepsy Experiments

A number of experiments on 27 classes were performed, varying T, α, and the keyword list. A limit to the improvement in accuracy resulted after some variation of these parameters. To gain better insight into the problem a number of experiments on nine classes were performed, again varying T, α, and the keyword list. The change of the keyword list allowed classification accuracies exceeding those achieved in any of the 27 class experiments. This clearly indicates that greater accuracy is possible over nine classes than 27 classes using the keyword set provided with the data base.

One final experiment was performed, experiment 13, again over the 27 classes and is indicated in Table 6.8. In this experiment, the a priori probabilities in the frequency table were generated for all the 1,751 keywords originally supplied with the data base, and this was done using all 500 documents. The resulting accuracy of 44.9% demonstrates that for this keyword set, the algorithm cannot adequately distinguish 27 classes for this Epilepsy data base. A subsequent improvement of these keywords is required.

CHAPTER 7

OVERVIEW OF THE SEQUENTIAL EXPERIMENTS

The experimental results of the last three chapters clearly demonstrate the validity of the sequential technique. Reasonable classification accuracy was achieved for each of the four different sets of documents studied. In order to achieve an improvement over initial classification results, each of the aspects of the classification technique was examined. These aspects are the sample set of documents, the keywords, the algorithm input parameters of T, R, and α, and finally the algorithm itself. This chapter evaluates the experimental results of the investigations into these aspects of sequential classification. This is done in order to summarize their effects on classification accuracy and to clarify the nature of their interdependencies.

7.1 Selection of Keywords

We found that more than any other parameter, changes in keyword sets were responsible for major fluctuations in classification accuracy. For the SPIN data base, more time was spent on improving keywords than for the other data bases; to a great extent this explains why better classification accuracy was obtained for the SPIN documents in the experimental results previously presented.

Keywords which intuitively seemed good often proved to be misleading. Since the sequential method assumes the independence of keywords, the effect created by the occurrence of keywords depends almost entirely on the individual class frequencies of each keyword. Hence it is not unlikely that the effect of a few good keywords may be nullified by the presence of a high frequency bad keyword - resulting in misclassification. The iterative "good-bad" keyword criterion proved effective for identifying those keywords which did not contribute to accurate classification. This experience motivated the development of the more

systematic Bayesian distance criterion for the detection of "noisy" or misleading keywords. This criterion is discussed in Chapters 8 and 9.

In Chapter 4, it was shown that a totally automatic keyword selection approach is feasible. For SPIN documents, an accuracy of 73% was obtained compared to a best classification accuracy of 81% achieved using keywords requiring manual intervention for their derivation. Further work should be done on automatic keyword selection algorithms.

In the comparison of the sequential classification approach to discriminant analysis given in Chapter 5, substantially fewer keywords were used for testing documents from the SPIN data base. The best results for the sequential algorithm were obtained using 67 keywords, whereas most SPIN experiments of Chapter 4 were conducted using 634 keywords. This magnitude reduction in the number of keywords shows that effective classification can be obtained with a small keyword list if that list is selected carefully. The price paid for this reduction is that a larger number of documents will contain few or no keywords at all, and must be manually classified by other means.

7.2 Experimentation with Parameter Values

The most important input parameter to be utilized in "tuning" the algorithm is α. In every data base, experimenting with α registered only moderate changes in classification accuracy. Parameter T was somewhat more effective in improving the classification accuracy of the sequential algorithm. In general, raising T improves accuracy by excluding a number of documents with few keywords from automatic classification, and in that sense, T is quite useful in instances where high classification accuracy is required. Human interpretation can then be allowed to manually classify those documents excluded by the T-value. The R parameter was not very effective in any of the experiments done in this study. The R parameter should be quite useful for full documents (in contrast to

abstracts) which contain many keywords. In such cases, R could tend to negate the effects of keyword proximity dependencies.

The discussion in Chapters 8 and 9 will show that the Bayesian distance can be viewed as a parameter of the sequential algorithm, and is considerably more effective in modifying the action of the algorithm than parameters α, T, or R. Some experiments along this line will be described in Chapter 9.

7.3 Experimentation With the Classification Algorithm

A number of modifications of the sequential algorithm were investigated. These variations included a change in the method by which α is calculated, and also the introduction of a ratio test. The change in the α calculation involved using appropriate a priori probabilities in the implementation of Bayes law. Although this had little effect on the classification accuracy in the cases examined, it does represent a logical extension which could be important in some other situation. The ratio test gave rise to the best sequential classification accuracy obtained in any experiments with the SPIN data base. Although the ratio test was somewhat ad hoc, it did lead to the more sophisticated Bayesian distance techniques for multiple classification described in Chapter 9.

As indicated in both Chapters 4 and 6, keyword stability studies were conducted to determine how gathering keyword frequencies over the entire data base rather than over a sample set changes classification accuracy. The results of these stability studies showed that both the SPIN and Epilepsy data bases were quite insensitive to gathering different keyword frequencies, whereas the accuracy for the Kucera-Francis data base increased from 52.8% to 68.7% when frequencies were obtained over all documents from this data base. This confirms the validity of the extensive experimentation on the SPIN data base, but shows why care must be taken to assure this stability in general.

7.4 Elements of Sequential Classification

Sequential classification depends upon interrelationships between three elements: documents, keywords, and classes. In general, the problems with documents are the problems associated with any natural language text. All of the experiments, but especially those comparing the sequential method with discriminant analysis, showed that the keyword sets were more important than the sample documents selected. The sample documents could be selected at random, and as long as they were sufficiently representative of the data base, the sequential algorithm operated satisfactorily. Discriminant analysis, on the other hand, required that the sample documents be carefully selected, or problems were encountered. This is fortunate from a practical point of view, for it is easier to select appropriate keywords than documents with certain characteristics.

Another aspect to the experiments conducted with the sequential algorithm is that various numbers of classes were associated with each study. It seems reasonable that accuracy may be inversely related to the number of classes involved. Table 7.1 shows this for a number of experiments on various data bases, and compares accuracy to that of randomly choosing the correct class. Keep in mind here, however, that the data bases are diverse, and also that considerably more work was spent on the keyword selection for SPIN than for the other data bases.

DATA BASE	NUMBER OF CLASSES	AV. CLASSIFICATION ACCURACY (%)	RANDOM CHANCE ACCURACY (%)
Solid State	3	83.4	33.3
SPIN	7	79.6	14.3
Epilepsy	9	53.6	11.1
SPIN	11	67.0	9.1
Kucera-Francis	16	40.0	6.2
Epilepsy	27	44.9	3.7

Table 7.1 Classification Accuracy Dependence on Number of Classes

7.5 Conclusions

The sequential method is a powerful and promising tool for high-speed classification and retrieval. In addition to its speed, the classification accuracy is competitive with that of the best methods published to date. Furthermore, the algorithm is highly stable and relatively independent of parametric or other algorithmic variations. Perhaps the most exciting work, however, is that investigating Bayesian distance, which is described in Chapters 8 and 9. The only truly "sensitive" element of the method as it now stands is the keyword set, but considerable work has been done on keyword selection techniques, and the modified algorithm of Chapter 9 will offer still another approach.

CHAPTER 8

THE BAYESIAN DISTANCE

The sequential method that has been described in the earlier chapters is a fast and fairly accurate method for document classification. It operates best under ideal conditions where a document that is being classified is well behaved. By this, we mean that the keywords contained in the document are concentrated primarily in one class, viz., the class in which the document actually belongs. In such a case, the a posteriori probability of the class to which the document belongs increases at each stage of the sequential process and the other classes gradually drop out from consideration until finally only the correct one remains. In a non-ideal case, however, documents may contain keywords which give a classification bias towards classes which are not the correct ones. Since no discrimination is made on the basis of the quality of keywords, inappropriate keywords are treated the same as good ones, and very often misclassification results.

This chapter introduces the concept of keyword vectors and shows how a document may be viewed as a point in an n-dimensional keyword space. A distance measure called the Bayesian distance is then defined on this space. Using this distance, noisy keywords can be detected and subsequently discarded from consideration in the classification algorithm. Finally, it is shown how this distance measure can be used to obtain a systematic method for multiple classification. Further elaboration and details of this study are given in reference [24].

8.1 Limitations of the Sequential Method

In order to understand some of the failings of the sequential method it is necessary to look at the most critical step of the algorithm, viz., the step that calculates the a posteriori probability of the classes. Given a string of keywords $W_1, W_2, \ldots, W_n$, the values for each class is calculated as follows:

$$\alpha_i = \frac{P(C_i)\ P(W_1|C_i)\ \cdots\cdots\ P(W_n|C_i)}{\sum_j P(C_j)\ P(W_1|C_j)\ \cdots\ P(W_n|C_j)}\ .$$

There are several salient points to note about this Bayesian technique. First, the keywords are treated as being completely independent, i.e., the following is assumed

$$P(W_1W_2\ \ldots\ W_n/C_i) = P(W_1/C_i)\cdot P(W_2/C_i)\ \ldots\ P(W_n/C_i),$$

whereas in fact a more rigorous and accurate technique would be to use the following

$$P(W_1,\ W_2,\ \ldots\ W_n/C_i) = P(W_1/C_i)\cdot P(W_2/C_i,\ W_1)\ \ldots\ P(W_n/C_i,\ W_1, W_2\ \ldots\ W_{n-1}).$$

It becomes clear from these two alternate approaches that an assumption of keyword independence reduces the computational complexity of the problem immensely. If independence is not assumed then bi-gram, tri-gram, and other frequencies have to be computed which would become very time consuming.

The second major point is that because of the assumed independence, every keyword's effect on the α-value for a particular class is directly proportional to its frequency of occurrence in that particular class. In other words, if we have three keywords of moderate frequencies f_1, f_2, f_3 from a class C_i and a fourth keyword of very high frequency f_4 from a class C_j such that $f_4 >> f_1 \cdot f_2 \cdot f_3$, then depending on the default probabilities used, chances are that class C_j will have a higher α-value than class C_i after the occurrence of the fourth keyword. This may or may not be a desirable situation, depending on which class the document actually belongs to.

From these two observations made above about the sequential method, two of its major drawbacks can be listed as follows:

First, the sequential method is unable to isolate inappropriate keywords from a set of good keywords. As in the example given above, the keyword with frequency f_4 may be a "noisy" keyword and should be recognized as such and not

considered for the probability calculations. It is also obvious that a decision about the appropriateness of this keyword has to be determined in the context of the three keywords which have previously occurred. Thus, even though we do not want to have to calculate keyword dependencies at each stage, we do need a measure of "closeness" between keywords. This measure should be able to isolate "noisy" keywords from good ones based on the distance between keywords. As will be shown later, the Bayesian distance measure that has been developed does this very effectively.

Second, the sequential method lacks the power to classify documents into more than one class in a systematic fashion. Very often, based on its subject content, a document should be classified in more than one class. The only way the sequential method can achieve this is by ranking the α-values in descending order and by assigning the document to classes in that order. Let us say that we have a three-class problem, i.e., an incoming document may be classified in classes C_1, C_2, C_3. At the point of classification let the a posteriori probabilities be given by α_1, α_2, α_3, where $\alpha_1 > \alpha_2 > \alpha_3$. Since the sequential algorithm has classified the document, it must be true that $\alpha_1 > \alpha$ and $\alpha_2 < \alpha$, where α is the preset threshold level. The primary class is, of course, C_1 but how about a secondary class? The best bet would be to obtain C_2 as a secondary class since $\alpha_2 > \alpha_3$. However, how do we decide whether C_2 or C_3 are preferable, or indeed whether a secondary class is justified at all? It may be that more of the document needs to be read before either C_2 or C_3 can be selected. We will show that some of the properties of the Bayesian distance can be used to provide answers to these questions. Before we introduce the concept of Bayesian distance, a brief discussion about the vector representation of keywords is necessary.

8.2 Vector Representation of Keywords

The starting point of the sequential method is essentially a matrix of conditional probabilities which give a numerical measure of the extent to which a keyword describes a particular class. The matrix may be specified as

$P(W_i/C_j)$ = Probability of keyword W_i occurring in a document which is known to belong to class C_j.

As the sequential method reads documents and extracts keywords, the only information it uses for a keyword is the set of associated probability values. A keyword can be looked upon as a k-dimensional vector of the form

$$W_i : [p_1, p_2, \ldots, p_j, \ldots, p_k]$$

where

$p_j = P(W_i/C_j)$, and k denotes the number of classes.

Two words, even though they are different, will have the same effect as far as the probability calculations are concerned if the vectors associated with them are the same. In other words, they can be treated as being mathematically synonymous. By the same token, two words with very different probability vectors may produce diverse effects in the a posteriori probability calculations. Besides giving an idea of the relationship between keywords, this view of keyword representation aids in the development of a very simple model for the representation of a document.

8.3 Vector Space Representation of Documents

A document for the purposes of classification is a string of keywords W_1, W_2, ..., W_n. Each of these keywords can be plotted in a k-dimensional space and the relationship between each can be studied. For the sake of simplicity, let us consider a 3-class case and a document that has a total of seven keywords in it. A plot of the document may appear as shown in Figure 8.1.

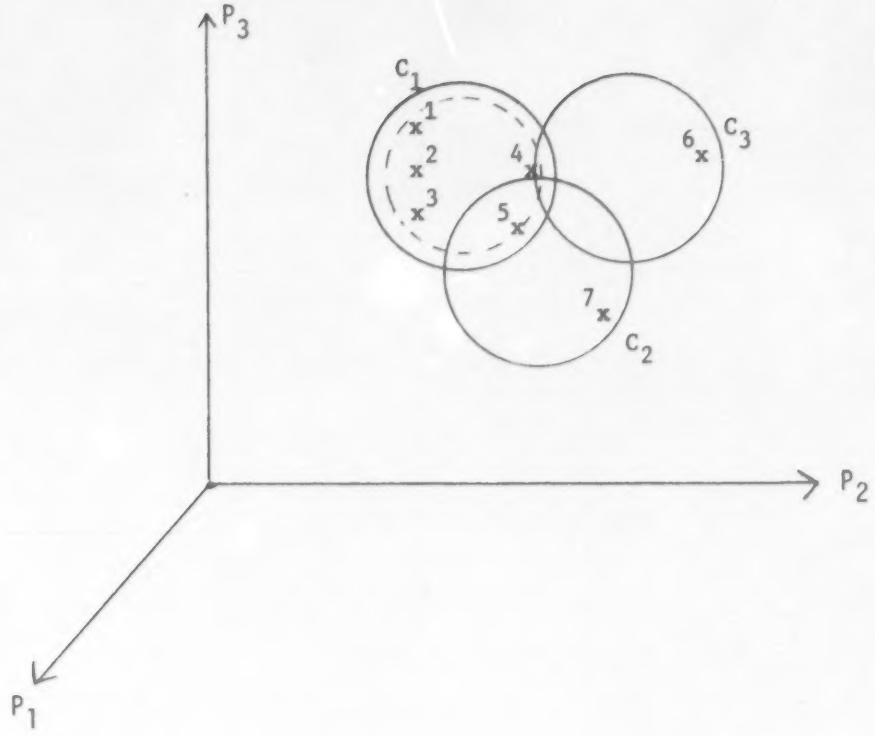


Figure 8.1 Vector Space Representation of a Document

We can say intuitively that the document belongs to class C_1 with the keywords 1, 2, 3, 4, and 5 totally or partially occurring from that class. These words which occur within the dotted boundary can be said to form a cluster of similar words. Assuming that the document does belong to class C_1, keywords 6 and 7 are definitely 'noisy' words and should be discarded. By defining a suitable distance measure on this space such clusters of good keywords and occurrences of noisy keywords may be detected.

Instead of having one single cluster of keywords, a plot of the given document may appear as in Figure 8.2. Here we notice that two distinct clusters of keywords occur. The first cluster might point to a primary class and the second one to a secondary class. In the next chapter we will demonstrate a technique using the Bayesian distance to detect the presence of such clusters and obtain multiple classification.

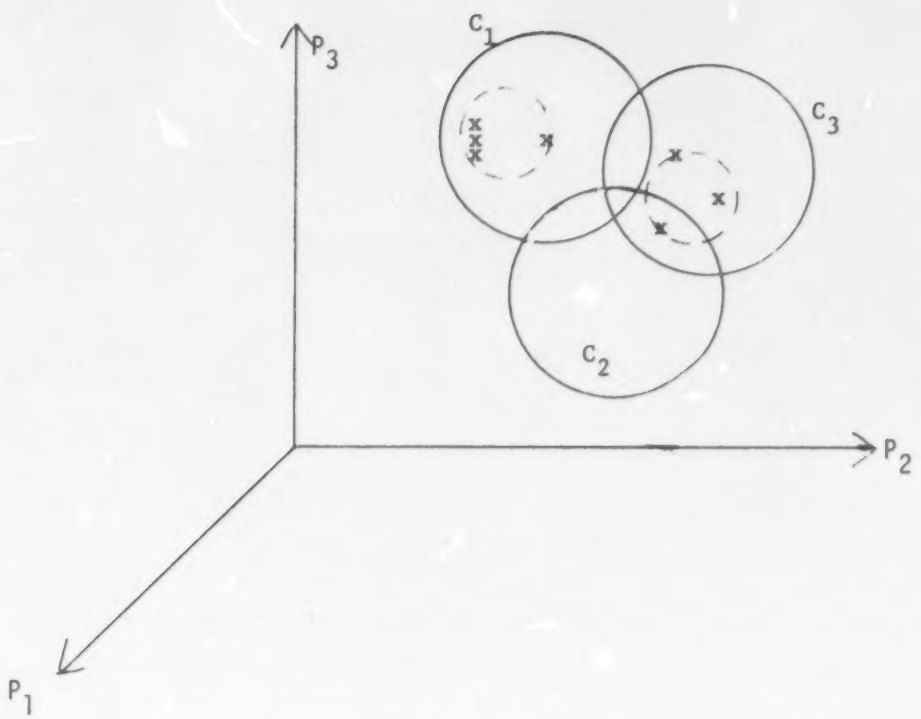


Figure 8.2 Clusters of Keywords

8.4 Definition of the Bayesian Distance

Consider k classes C_1, C_2, ..., C_k into which an incoming document may be classified. At a given stage of the sequential process suppose we have read n keywords W_1, W_2, ..., W_n, and that we represent the effect of all these keywords by a single variable y. Then the a posteriori probability of a given class after observation y has been made can be represented as follows:

$$P(C/y) = [P(C_1/y), \ldots, P(C_m/y)].$$

This will be called the Bayesian probability vector.

The Bayesian distance on the probability space of a set of classes after an observation y, denoted by $D_B(y)$, is represented by a magnitude and a direction, i.e., $D_B(y)$ = [Mag,Dir], where Mag is equal to the squared Euclidian norm of the Bayesian probability vector and Dir is the index of the class having the highest

a posteriori probability after observation y,

$$Mag = \sum_{k=1}^{m} [P(C_k/y)]^2, \qquad (8\text{-}1)$$

and Dir = i such that

$$P(C_i/y) = \max_k [P(C_k/y)], \quad k=1 \text{ to } m.$$

8.5 Variation of Bayesian Distance With Keywords

In order to test the validity of the concept of Bayesian distance and its performance with an actual data base, a set of experiments was conducted using the SPIN document set. The purpose of the experiments was to note the variation of the magnitude and direction of the Bayesian distance with each keyword read from a given document.

Each document was read entirely and a list of all the keywords present was formed. Let this list be represented by

$$\{W_1, W_2, \ldots, W_n\}$$

where W_i is the i^{th} keyword occurring in the document. Each keyword W_i is a member of the keyword set and has an associated probability vector of the form

$$W_i: \ [p_1^i, p_2^i, \ldots, p_k^i].$$

Let α_j^i represent the a posteriori probability of class C_j after the i^{th} keyword. In Chapter 3, a method for its calculation was discussed. The Bayesian distance (D_B) magnitude and direction are calculated using these α_j^i values. Let Mag(i) denote the magnitude and Dir(i) denote the direction of D_B after the i^{th} keyword W_i is processed. Then

$$Mag(i) = \sum_{j=1}^{k} (\alpha_j^i)^2, \qquad (8\text{-}2)$$

with Dir(i) the index of the highest α_j^i value. Mag(i) and Dir(i) are calculated for each value of i ranging from 1 to n. Before the results of the experiments are reported several terms need to be defined.

Good Keyword: During the process of classification a keyword is considered to be a good keyword if its highest probability component belongs to the class to which the document belongs. In this case the keyword relates the document to the indicated class more highly than to any other class.

Noisy Keyword: During the process of classification a keyword is considered to be a noisy keyword if its highest probability component points to a class other than a class to which the document belongs.

Ideal Document: A document is considered to be ideal if every keyword contained in it is good.

Non-Ideal Document: A document is considered to be non-ideal if there is at least one keyword in it which is noisy.

It should be noted here that these definitions presuppose that the class to which a document belongs is known. During classification, however, the class membership of a document is not known until the end of the sequential process. A keyword that is extracted from a document at each step of this process is tagged as either good or noisy. This decision depends only on the keywords that have preceded this particular keyword, and may change as more of the document is examined. This point is clarified later in the chapter.

8.6 Analysis of Ideal Documents

Twenty documents were selected from the set of ideal documents for detailed study. In all cases the D_B magnitude increases with the number of keywords read until it reaches or closely approaches a value of unity, the direction being constant over the entire range. This phenomenon is illustrated in Figure 8.3. This fact can be utilized to recognize documents which are ideal in nature and classify them into a class designated by the direction of D_B. The fact that such a situation can also occur for non-ideal documents will be discussed later.

The nature of the variation of the magnitude of D_B has been utilized to design a classifier which searches for such a pattern in the Bayesian distances obtained by processing the keywords in a test document. Figure 8.3 illustrates how the D_B curve gradually approaches the value of unity over a range of keywords. The magnitude of D_B at any stage of the sequential process depends on the set of α_j^i values $\{\alpha_1^i, \alpha_2^i, \ldots, \alpha_k^i\}$ computed at that stage. The only way that the magnitude can assume a value of unity is when one of these α_j^i is unity and the rest are all zero. Suppose n keywords $\{W_1, W_2, \ldots, W_n\}$ have been read. Without loss of generality, let us assume that each keyword W_i has an associated probability vector of the form $[p_1^i, d, \ldots, d]$, where d is the default probability value. Hence, each of these keywords is strongly indicative of class C_1. If the α_j^i values are computed using these n keywords then the value of α_1^i is given by

$$\alpha_1^i = \frac{\prod_{i=1}^{n} (p_1^i)}{\prod_{i=1}^{n} (p_1^i) + (k-1)d^n} . \qquad (8\text{-}2)$$

Each of the other α_j^i values is given by

$$\alpha_j^i = \frac{d^n}{\prod_{i=1}^{n} (p_1^i) + (k-1)d^n}, \quad j > 1. \qquad (8\text{-}3)$$

Because $p_1^i >> d$, as more and more keywords of the same form are processed, α_1^i approaches unity and the other α_j^i approach zero. Theoretically, however, α_1^i never reaches a value of unity. The magnitude of D_B which is given by

$$\text{Mag}(i) = \sum_{j=1}^{K} (\alpha_j^i)^2$$

approaches a value of unity as α_1^i approaches unity. Therefore, Mag(i) can be made as close to unity as desired by processing more such good keywords. In practice this is not feasible because documents have a limited number of keywords. Besides, if a large number of keywords is read in order to make Bayesian distance

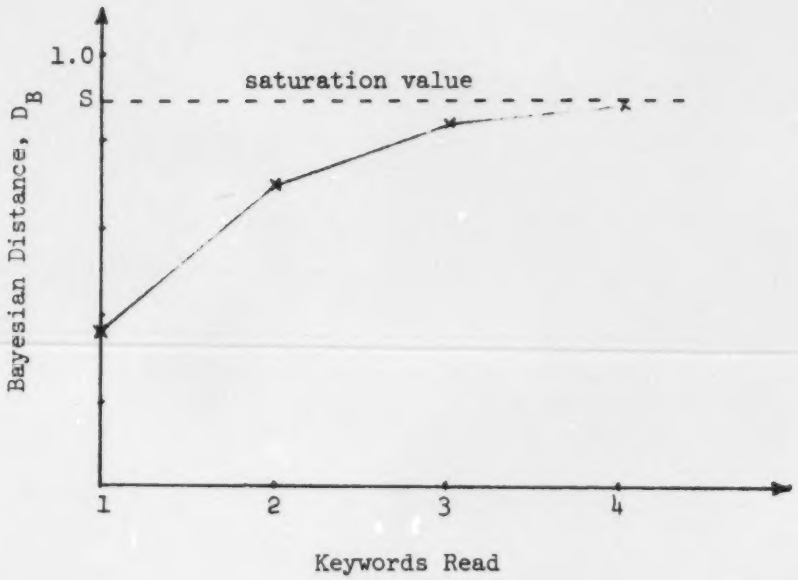


Figure 8.3 Bayesian Distance for Ideal Documents

magnitude very close to unity, the entire purpose of a sequential technique has been lost.

A practical implementation therefore requires the definition of two parameters. One is the number of keywords that need to be examined before a decision can be made. This parameter will be denoted by Q. The other is a parameter called the saturation value and will be denoted by S. In order that a document may be classified, the magnitude of D_B after Q good keywords are processed should be greater than or equal to S, so S represents a confidence level. The higher the value of S, the greater is the confidence that the document belongs to the class given by the index of D_B.

These two parameters can be determined experimentally for a given data base by taking samples of ideal documents and analyzing their Bayesian distance patterns. Such experiments have been conducted for the SPIN data base and the results are presented in Table 8.1 for the 20 ideal documents selected for detailed study. Before the results are analyzed, however, another important statistic should be mentioned, i.e., the initial value of the Bayesian distance. Let us denote this by B_I. This initial value B_I reflects the quality of the first keyword; for an ideal document all of whose keywords have predominantly high probability values for a single class, the higher the B_I value the more rapidly will the Bayesian distance reach the saturation threshold. In a practical implementation, the parameter Q can be changed based on the value of B_I. If B_I is low, Q can be increased because more keywords will be required to reach the saturation threshold. This is clarified in the discussion which follows.

The documents in Table 8.1 have been listed in ascending order according to the value of B_I. The third column in this table gives the number of keywords that are read before the magnitude of the D_B exceeds or equals the value of the parameter S. For the results given in Table 8.1, a saturation value of 0.99 was assumed. As can be seen from the table, a higher value of B_I requires, in

Table 8.1 Analysis of Ideal Document

SATURATION VALUE = 0.99

DOCUMENT NUMBER	INITIAL VALUE OF D_B (B_I)	NO. OF KEYWORDS REQUIRED FOR SATURATION	SPIN CATEGORY
1	0.353	5	S
2	0.389	3	M
3	0.522	3	M
4	0.522	4	M
5	0.562	3	M
6	0.636	3	S
7	0.648	3	M
8	0.649	3	M
9	0.650	3	S
10	0.737	2	P
11	0.757	3	D
12	0.798	2	T
13	0.923	3	S
14	0.923	2	S
15	0.953	2	S
16	0.967	2	M
17	0.970	3	S
18	0.983	2	T
19	0.985	2	P
20	0.995	1	P

general, that a greater number of keywords be read before saturation is obtained. In other words, the parameter Q should vary according to the value of B_I. In order to study the variation of this parameter, the range of B_I for the ideal documents was divided into four intervals. Experiments were conducted to obtain an average number of keywords required to reach saturation in each of these intervals. The results presented in Table 8.2 show that when B_I is lower, a higher average number of keywords are read before the saturation value is reached. During classification therefore, B_I can be used to determine the value of the parameter Q.

8.7 Analysis of Non-Ideal Documents

As in the case of ideal documents, 20 non-ideal documents were selected at random and analyzed. These documents are such that each one of them contains at least one noisy keyword. For clarity of presentation, only a sample set of the results obtained is shown in Table 8.3. In the columns of Table 8.3, the first value represents the magnitude of D_B and the second value denotes its direction. The direction corresponds to a numerical index assigned to each of the SPIN categories. The noisy keywords are underlined for ease in identification. As can be seen from the data, the occurrence of a noisy keyword is marked by a change in magnitude or direction or both in the Bayesian distance values. The general nature of this variation can be studied by identifying two cases depending upon whether only the magnitude changes or both magnitude and direction change.

This first type is shown in Figure 8.4. In this case the occurrence of a noisy keyword is marked by a decrease in magnitude, the direction remaining the same. After the noisy keyword, the magnitude again increases starting from the following keyword and reaches saturation as in the case of an ideal document. This situation is depicted by documents 1 and 4 in Table 8.3. The second keyword

Table 8.2 Experiments on Starting Intervals

SATURATION VALUE = 0.99

Intervals for B_I:

I_1: (0.3,0.5)

I_2: (0.5,0.7)

I_3: (0.7,0.9)

I_4: (0.9,1.0)

STARTING INTERVAL	NO. OF DOCUMENTS IN INTERVAL	AVERAGE NO. OF KEYWORDS REQUIRED FOR SATURATION
I_1	2	4.0
I_2	7	3.0
I_3	3	2.3
I_4	8	2.2

Table 8.3 Analysis of Noisy Documents

Data Given: Mag(i),Dir(i)

DOCUMENT NUMBER	KEYWORDS							SPIN CLASS
	1	2	3	4	5	6	7	
1	.880,3	.803,3	.999,3	.999,3				P (3)
2	.502,1	.500,1	.532,0	.976,1	.999,1			D (1)
3	.961,3	.675,3	.888,1	.903,3	.892,3			P (3)
4	.945,1	.481,1	.947,1	.999,1	.853,1	.998,1		D (1)
5	.389,2	.809,4	.430,3	.501,2	.438,4	.986,2	.999,2	M (2)

Note: Noisy keywords are underlined.

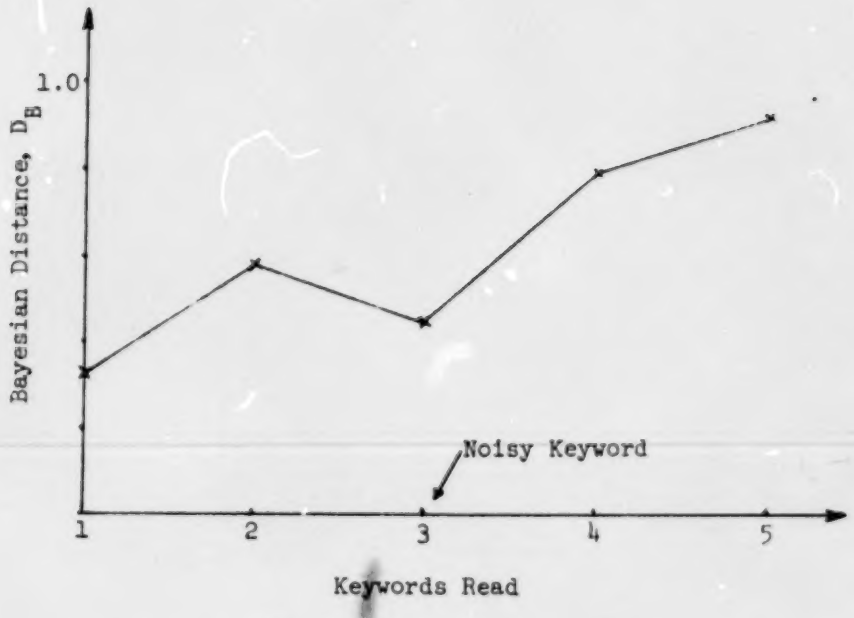


Figure 8.4 Change in Magnitude of Bayesian Distance

in document i is a noisy keyword and hence the magnitude drops from 0.880 to 0.803. When the third and the fourth keywords are processed, the magnitude increases from 0.803 to 0.999, and can both be identified as good keywords. For document 4, the same phenomenon is observed except that the second and fifth keywords are both noisy.

Another effect caused by the occurrence of a noisy keyword is the change in the direction of the Bayesian distance. Here the noisy keywords are so strongly biased towards a wrong class that a change in direction occurs irrespective of the nature of the change in magnitude. This situation is depicted by documents 2, 3, and 5 in Table 8.3. In document 3, the third keyword is very strongly indicative of class C_1. As a result even though the first keyword was strongly indicative of class C_3, the direction changes after the third keyword is processed. It can therefore be isolated as a noisy keyword in relation to the first keyword. In document 5, after the second keyword is processed, the direction changes class C_2 to C_4. Hence it is considered to be noisy in relation to the first keyword. The difference between this case and the situation depicted by document 3 is that here the first keyword was not very strongly indicative of class C_2. Hence a change in direction from $C_2$to C_4 could have occurred even if the second keyword were not very strongly indicative of class C_4.

A classifier may be designed which will detect such variations in magnitude or direction, and extract the keywords responsible for causing them. These keywords may then be identified as noisy in relation to the class currently under consideration and discarded. An efficient procedure for implementing such a technique will be discussed in the next chapter.

CHAPTER 9

THE REVISED SEQUENTIAL METHOD

The previous chapter showed how the Bayesian distance measure could be used to detect the presence of noisy keywords and to identify groups of good keywords in a document. The basic sequential algorithm discussed in Chapter 3 can now be modified to achieve classification by taking into account only the good keywords. This chapter is devoted to the design of such an algorithm. The algorithm works in two phases. In the first phase keywords are extracted sequentially from a document and, based on the Bayesian distance analysis, the total number of keywords read are divided into two groups -- the good keywords and the noisy keywords. The first section of this chapter discusses a method which achieves such a separation of the keywords. When the good keywords are such as to meet the Q and S thresholds discussed in the previous chapter, the algorithm analyzes these words to obtain a primary class for the document as the second phase.

9.1 Separation of Good and Noisy Keywords

Each keyword extracted from a document has the form W_j: $[p_1, p_2, \ldots p_k]$ where p_i denotes the probability $p(W_j/C_i)$. Besides a probability vector of the form shown above, each keyword has also associated with it an index I, representing its location in the document. The first keyword read from a document has an index 1, the second keyword has an index 2, and so on. Each stage of the sequential process now consists of the following.

i) Q keywords are read from a document where the parameter Q represents a threshold. The D_B magnitude has to increase monotonically over a sequence of Q keywords in order for a document to be classified in a primary class. This parameter was introduced and discussed in Chapter 8.

ii) The indices and the associated probability vectors for each of these Q keywords are stored in an input buffer.

In order to separate the noisy keywords from the good ones, two auxiliary buffers are set up. Each of these buffers are capable of storing the index of a keyword which serves as a pointer to an entry in the input buffer. The first auxiliary buffer will be used to store the indices of the good keywords, and will be called the good keyword buffer (GBUF). The second auxiliary buffer will be used to contain the indices of the noisy keywords, and will be referred to as the noisy keyword buffer (NBUF).

Initially let us assume that all the Q keywords read at any stage of the sequential process are good keywords in that they are all indicative of one class. Therefore, the indices of these Q keywords are tentatively loaded in the GBUF. Suppose $W_1, W_2, \ldots, W_Q$ are the Q keywords whose indices are in the GBUF. In order to check whether a monotonically increasing pattern in the magnitude of D_B is obtained over these Q keywords, the Bayesian distances are calculated using keyword W_1, then keywords W_1 and W_2, and so on. The form of such a monotonically increasing pattern was shown in Figure 8.3 of the previous chapter. If at any stage a particular keyword causes the magnitude to decrease or the direction of D_B to change, it is tagged as a noisy word. Its index is then removed from the GBUF and put in the NBUF. For example, suppose four keywords W_1, W_2, W_3, and W_4 have been read. The indices 1, 2, 3, and 4 are tentatively loaded into the GBUF. Further suppose that keywords W_1 and W_2 yield a monotonically increasing pattern in the magnitude of the Bayesian distance and a constant direction. It is then assumed that they are good keywords in relation to each other. Now let keyword W_3 be such that it causes a decrease in the magnitude. Then it is tentatively tagged as a noisy keyword and the index 3 is removed from the GBUF and put in the NBUF. The situation at this stage is depicted by Figure 9.1. The process of calculating the Bayesian distance is repeated using keywords W_1,

INPUT BUFFER

INDEX	KEYWORD	PROBABILITIES			
		1	2	. . . .	k
1	W_1	p_1	p_2	. . . .	p_k
2	W_2	q_1	q_2	. . . .	q_k
3	W_3	r_1	r_2	. . . .	r_k
4	W_4	s_1	s_2	. . . .	s_k

GBUF

1	2	4	. . . . .

NBUF

3	. . . . .

Figure 9.1 Input Buffer, Good Keyword Buffer and Noisy Keyword Buffer

W_2, and W_4 to check whether W_4 is a good or a noisy keyword. The following section describes how the Bayesian distances are calculated and how the magnitudes and directions are tested at each stage to identify the noisy keywords.

9.2 The Bayesian Distance Calculator and Noise Detector

Let Mag(i) and Dir(i) denote the magnitudes and directions respectively, calculated at the end of the i^{th} keyword in the GBUF. This is done by first calculating the α_j values using the first i keywords in the GBUF.

$$\alpha_j = \frac{\prod_{r=1}^{i} p(W_r/C_j)}{\sum_{j=1}^{k} \left[\prod_{r=1}^{i} p(W_r/C_j) \right]}, \quad j = 1 \text{ to } k. \tag{6-1}$$

Mag(i) and Dir(i) are then calculated using these values as follows:

$$\text{Mag}(i) = \sum_{j=1}^{k} (\alpha_j)^2 \tag{6-2}$$

Dir(i) = index of the highest α_j value.

For each value of i greater than two, a noise detection procedure is implemented. If either of the following two conditions

i) Mag(i) < Mag(i-1), or

ii) Dir(i) ≠ Dir(i-1),

is detected then the i^{th} keyword in the GBUF is assumed to be noisy. Its index is put in the NBUF and removed from the GBUF. The index of the next keyword in the input buffer is then loaded into the GBUF and the process is repeated.

When the following conditions are found,

i) Mag(1) < Mag(2) < ... < Mag(Q),

ii) Mag(Q) ≥ S where S is the preset saturation value discussed in Section 8.6, and

iii) Dir(1) = Dir(2) = ... = Dir(Q),

then a monotonically increasing pattern in the magnitudes of the Bayesian distances, with the direction remaining constant, is obtained. At this point there are Q keywords in the GBUF which are good keywords in relation to each other and which between them indicate a unique class. This class, whose index is given by Dir(Q), is identified as the primary class for the document. In case one of the three conditions listed above is not satisfied and the document contains no more keywords to be read, then it is termed unclassifiable.

It has been shown that after reading each keyword a decision is made as to whether it should be considered good or noisy depending on the direction and magnitude of the Bayesian distance. Based on this decision, the index of the word is either kept in the GBUF or is put in the NBUF. Since a keyword is good or noisy in relation to the other keywords that have preceded it in the document, it is quite possible that at any stage of the sequential process, the NBUF contains the good keywords which point to the correct primary class. This can be better illustrated by an example.

Suppose we have read a series of keywords W_1, W_2, W_3, W_4, and W_5. Then let us assume that the following sequence of actions have been taken:

i) the index of W_1 is put in the GBUF,

ii) W_2 is good in relation to W_1 and so its index is put in the GBUF, and

iii) W_3, W_4, and W_5 are noisy in relation to W_1 and W_2, and so their indices are put in the NBUF.

At this point the GBUF contains the indices of W_1 and W_2, while the NBUF contains the indices of W_3, W_4, and W_5. Suppose W_1 and W_2 are indicative of class C_1 and W_3, W_4, and W_5 are all indicative of class C_j. If now the magnitude of D_B calculated by using W_3, W_4, and W_5 exceeds the value of the magnitude calculated by using W_1 and W_2 in the GBUF, it is highly likely that W_3, W_4, and W_5 are the set of good keywords which point to the correct primary class. If this is the

case, then the indices of the words in the GBUF should be interchanged with those in the NBUF. To achieve this, at every stage when the number of words in the NBUF equals that in the GBUF, the Bayesian distances of the words in it are calculated. If a unique direction is obtained and if the magnitude exceeds the magnitude of the Bayesian distance calculated by using the words in the GBUF then the two buffers are interchanged.

The concepts discussed in this section have been implemented as a primary classification algorithm. The next section briefly discusses this algorithm.

9.3 Description of the Revised Sequential Algorithm

The first phase of the classification algorithm which obtains a primary class for a test document contains several parts, the most important of which are the primary classifier, the Bayesian distance calculator and the noise detector. In this section, we will briefly discuss these components of the algorithm. The portion of the algorithm which extracts keywords from a document will be described for the sake of completeness. A flowchart of the algorithm is given in Figure 9.2.

9.3.1 The Keyword Extractor

This portion of the algorithm reads one keyword at a time from a given test document and stores the following in an input buffer:

i) a value I corresponding to the index of the keyword read,

ii) the keyword, and

iii) the probability values associated with the keywords. These are stored in a matrix P(I,j), where I represents the index of the keyword and j represents the index of a category.

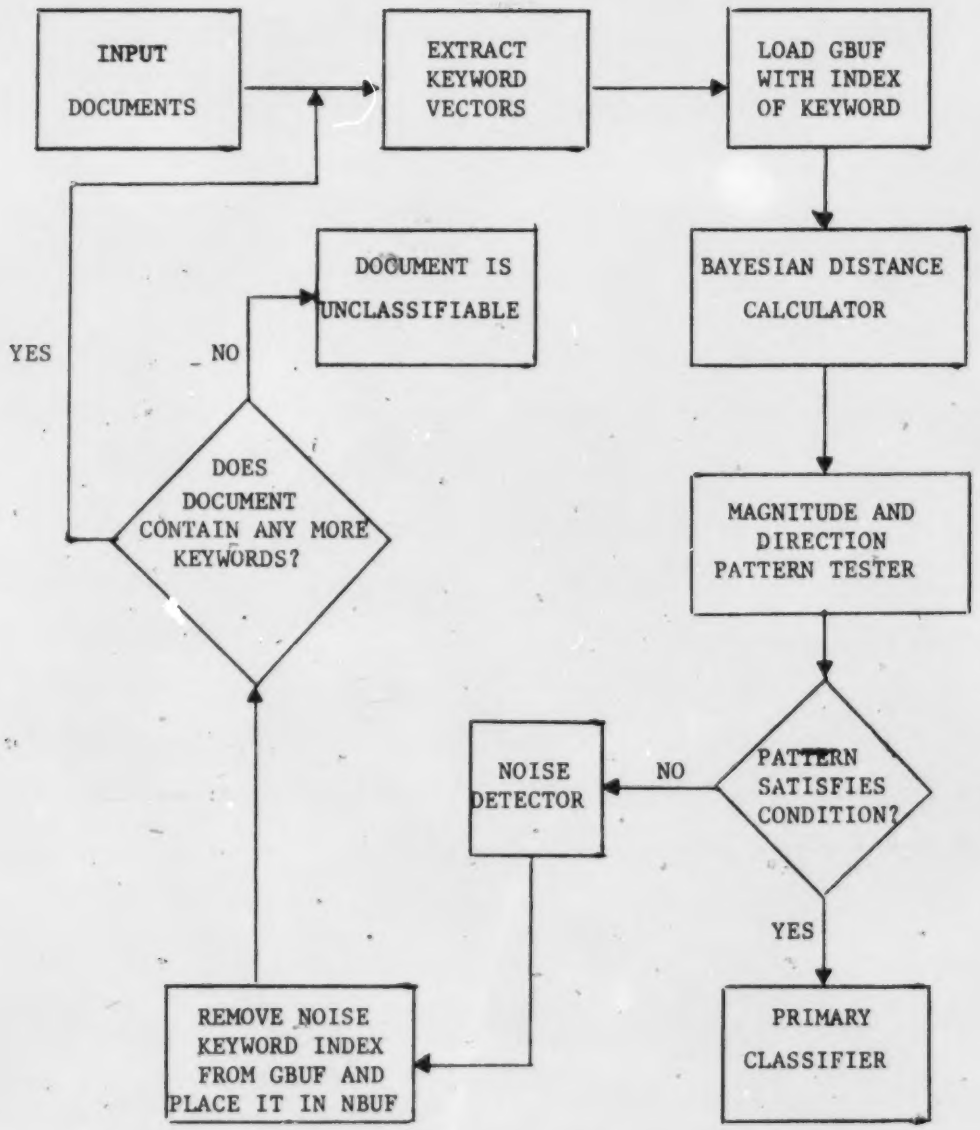


Figure 9.2 Flowchart of the Revised Sequential Algorithm

9.3.2 The Bayesian Distance Calculator

The input to this portion of the algorithm consists of a set of indices S_T, which comprise either the set of indices of the keywords in the GBUF or the set of indices of the keywords in the NBUF. Let this set S_T at any given stage be $S_T = \{i_1, i_2, \ldots, i_r\}$. Then the Bayesian distance calculator computes the magnitudes and directions of D_B considering the keywords W_{i_1}, then W_{i_1} and W_{i_2}, then W_{i_1}, W_{i_2}, and W_{i_3}, and so on. The magnitudes and directions $Mag(i_1)$, $Mag(i_2)$, ..., $Mag(i_r)$ and $Dir(i_1)$, $Dir(i_2)$, ..., $Dir(i_r)$ are stored in an array for analysis by the noise detector.

9.3.3 The Noise Detector

Using the array of magnitudes and directions determined by the Bayesian distance calculator the noise detector checks for a monotonically increasing pattern in the magnitudes. If this is satisfied, it checks to see whether the directions are all the same. Suppose in the array of magnitudes and directions one or both of the following two conditions

i) $Mag(i) < Mag(i-1)$

ii) $Dir(i) \neq Dir(i-1)$

is detected, then the i^{th} keyword is identified as a noisy keyword.

9.3.4 The Revised Sequential Algorithm

The input to this section of the algorithm consists of the array of magnitudes and directions calculated by the Bayesian distance calculator and the indices of the noisy keywords computed by the noise detector. The parameters Q and S, discussed earlier, govern the operation of this section of the algorithm. The primary classifier performs the following functions.

i) It loads the set S_T with a set of Q indices from the GBUF and uses the Bayesian distance calculator to compute an array of magnitudes and directions.

ii) It uses the noise detector to identify a noisy keyword in the set of Q keywords obtained from the GBUF.

iii) If the noise detector identifies a noisy keyword, it removes the index of this keyword from the GBUF and places it in the NBUF. It then uses the keyword extractor algorithm to obtain an additional keyword. If there is such a keyword, its index is loaded into the GBUF. If there are no more keywords in the document then it identifies the document as being unclassifiable.

iv) Each time the index of a word is loaded into the NBUF, it checks to see whether the size of the NBUF equals that of the GBUF. If so, then it checks to see whether, based on the Bayesian distance values, the contents of the NBUF should be interchanged with those of the GBUF. The criterion which governs such an action has been discussed in Section 9.2.

v) If the noise detector does not identify a noisy keyword in the Q keywords present in the GBUF, the primary classifier checks to see whether $Mag(Q) \geq S$ where S is the saturation value. If so, then the class given by Dir(Q) is identified as the primary class for the document. If $Mag(Q) < S$, then an additional keyword is read. If the document does not contain any more keywords, the primary classifier still classifies it into the class given by Dir(Q) but records the fact that the S value was not satisfied. This is to signify that the confidence in the primary class obtained for this document is less than those which have satisfied the S threshold.

This algorithm has been implemented to classify approximately 500 documents contained in one of the releases of the SPIN data base. A brief description of this data was given in Chapter 4. The following section presents the results that have been obtained.

9.4 Implementation of the Algorithm and Results of Experiments

Since each of the SPIN documents has been preclassified by the American Institute of Physics (AIP), an estimate of how well the algorithm has performed could be easily obtained. This was done by comparing the classification obtained by the algorithm with that given by the AIP.

As pointed out in Section 9.3, the algorithm has two parameters Q and S. The parameter S denotes the final value of the Bayesian distance at the point when a document is classified. It represents a measure of confidence in the classification that is obtained. The parameter Q represents the number of keywords over which the Bayesian distance has to satisfy a monotonically increasing pattern in the magnitude, while maintaining a constant direction. Initially S was varied from 0.8 to 1.0 for different values of Q, viz., Q = 2, 3, and 4. In all these cases it was found that if a document satisfied the Q threshold, then the final value of the Bayesian distance, i.e., Mag(Q) always resided in the interval 0.9 to 1.0. Variation of S, therefore, did not produce any significant change in the results and hence for all subsequent experiments a fixed value of S = 0.9 was used. Before discussing the results of these experiments further, several terms need to be defined. Let

D = number of documents correctly classified

T = total number of documents in the data base

V = number of documents found unclassifiable.

Then

A_1 = accuracy over those classified = $\frac{D}{T-V} \times 100$

A_2 = overall accuracy = $\frac{D}{T} \times 100$.

The first set of experiments conducted used a fixed value of Q. For values of Q = 2,3, and 4, the results are presented in Table 9.1. As can be expected, an increase in Q results in higher values for A_1. This is because as

Table 9.1 Primary Classification with Fixed Q

T = 500

Q	NUMBER OF DOCUMENTS CORRECTLY CLASSIFIED D	NUMBER OF DOCUMENTS INCORRECTLY CLASSIFIED T - (D + V)	NUMBER OF DOCUMENTS FOUND UN-CLASSIFIABLE V	ACCURACY OVER THOSE CLASSIFIED A_1	OVERALL ACCURACY A_2
2	326	116	57	73.9%	65.2%
3	340	62	97	84.6%	69.7%
4	301	51	147	85.5%	61.7%

Q increases more keywords are examined before a document is classified. Thus, for Q = 2 only two good keywords are needed before a primary class is identified. In many cases this might lead to a precipitous decision. Keywords occurring in the rest of the document may point to an entirely different class which in many cases may be the correct class. An increase in Q avoids such precipitous decisions and hence increases the value of A_1.

Hoever, an increase in Q means a document has to have a greater number of good keywords in order to be classified. Fewer documents are able to satisfy this more stringent criterion. Hence the number of unclassified documents increases and the value of overall accuracy (A_2) decreases.

Some of the unclassified documents are such that they contain a set of good keywords indicative of a unique class. They are identified as unclassifiable by the algorithm because the number of good keywords contained in them is not high enough to satisfy a fixed Q threshold. Therefore in the next set of experiments, the value of Q was dynamically varied during classification. Table 9.2 presents the results of these experiments. If a particular document did not satisfy a given Q threshold then the value of Q was reduced by one and the document was reconsidered for classification under the following conditions:

i) the document has been read entirely, and

ii) no noisy keyword has been identified in the entire document.

For example, suppose the starting value of Q is three, and a test document contains only two keywords, both of which are good keywords in relation to each other, i.e., they are both indicative of one class. Then the value of Q would be reduced to two. The primary classifier would then attempt to identify a primary class for the document based on these two keywords.

The results presented in Table 9.2 indicate that some of the previously unclassified documents are now correctly classified. Therefore, the value of overall accuracy increases. However, because of the reduction in Q, some of the

Table 9.2 Primary Classification with Dynamic Q

T = 500

RANGE OF Q	NUMBER OF DOCUMENTS CORRECTLY CLASSIFIED D	NUMBER OF DOCUMENTS INCORRECTLY CLASSIFIED T - (D + V)	NUMBER OF DOCUMENTS FOUND UN-CLASSIFIABLE V	ACCURACY OVER THOSE CLASSIFIED A_1	OVERALL ACCURACY A_2
1-3	380	105	14	78.4%	77.9%
1-4	386	71	42	84.5%	79.1%

previously unclassified documents are now incorrectly classified. Therefore, the value of A_1 tends to decrease.

9.5 Secondary Classification

The discussion in this chapter has dealt with the design and implementation of an algorithm which obtains primary classes for a set of documents. A section of this algorithm separates the noisy keywords from the good keywords in a given document. The noisy keywords are put in a separate buffer called the NBUF. In effect this means that at the end of primary classification two different groups of keywords are obtained. The good keyword group is analyzed to obtain a primary class. But how about the group of keywords that have been tagged as noisy? If the contents of a document are such that a secondary class could be identified, then it is very likely that the keywords that have been isolated as noisy may actually be indicative of such a class. If this is the case, then the words in the NBUF could be analyzed to obtain a possible secondary class. Reference [24] addresses this problem and extends the revised sequential method to obtain secondary classification.

CHAPTER 10

CONCLUSIONS AND FUTURE WORK

This research has indicated the feasibility of the proposed sequential method for the automatic classification of documents. One of the main advantages of this approach is that the document to be classified need not be examined in its entirety before a decision regarding its class membership can be made. This and several other features of the sequential algorithm lead to a fast and efficient method, but this study has shown it has reasonably low misclassification error when applied to diverse types of document collections.

This chapter will highlight some of the major results and contributions of this study. In addition, some indication of future research problems recommended for further investigation will be given.

10.1 Conclusions of This Study

The sequential classification method has proved effective when applied to diverse data bases. As indicated in Section 3.2, this was done under worst case conditions, which if modified, should result in further reduced misclassification error. Over all data bases, it was found that the sequential method was most sensitive to changes in the keyword sets, so a thorough study should be made to obtain the best possible keyword set in applying the sequential algorithm to a particular data base. The sample documents are relatively unimportant, as long as they are sufficiently representative of the data base.

The best classification accuracies were obtained with abstracts of the SPIN data base. This is to be expected, however, since a major effort was made to obtain a best keyword set. The Kucera-Francis data base consisted of full text documents, and thus posed a different type of test for the sequential algorithm. The algorithm performed well, except when documents were presented

whose subject content was not homogeneous. This was the only data base in which keyword frequencies appeared unstable, as considerably improved classification resulted when frequencies were obtained over the complete document collection rather than just over the sample documents. The Epilepsy data base was investigated primarily to evaluate the keywords provided with the documents; the conclusion here is that this keyword set should be further improved for effective classification or retrieval.

A fairly definitive study was conducted in the comparison of the sequential algorithm with the discriminant analysis method of Williams [42,43]. A comparison was made using both SPIN documents and a solid state data base used by Williams, and the sequential algorithm performance was shown to be comparable if not superior. A surprising result of this comparative study was that sequential classification is still accurate with a markedly reduced keyword set, but the price paid is that fewer documents contained the required number of keywords.

A number of keyword extraction techniques were developed. It was shown that keywords could be generated automatically if sufficient care is taken for the data base under study. The iterative "good-bad" criterion was shown effective for modifying initial keyword sets. Although this method is somewhat ad hoc, it lead to the Bayesian distance criterion, which should prove to be a much more systematic approach to identify keywords which are too often misleading. An application of clustering methods for studying keywords and classes was shown effective in Section 4.3.4. This required the comparison of two different classifications, a difficult problem. In Section 2.2.3, a new method is proposed for this problem, applicable even for the complicated case of a classification with overlapping classes.

Possibly the most important results of this research was the development of the Bayesian distance criterion. In Chapter 8, two problems with the sequential

method were identified:

1) the sequential method is unable to isolate inappropriate keywords from a set of good keywords, and
2) the sequential method offers no systematic way to associate a document with multiple classes.

The Bayesian distance criterion was applied in Chapter 9 in the design of the revised sequential method, which not only obtained improved classification accuracy, but offers a systematic resolution of these two problems. This is an area in which further research is needed to fully develop this approach to its full potential.

10.2 Future Work

This research on the sequential algorithm has shown that more work is needed to develop systematic keyword selection procedures. Some general ideas are presented in Chapter 2, and the experiments in Chapter 4 show that automatic keyword selection is feasible. Perhaps the Bayesian distance criterion can be used in some way for the selection of keywords; presently it is more clear how this criterion can be applied to modify an existing keyword set.

An important practical consideration is how to utilize the sequential technique in a user-oriented retrieval environment. The primary advantage this classification has to offer in a real-time retrieval situation is its speed. One possible way this might be applied is to process user queries, and the degree to which these queries have been satisfied or not satisfied is utilized to gradually change the classification, so that improved user retrieval could be obtained in subsequent requests. The classification could then be slowly modified so as to best represent the user's view of the data base. The sequential method would be instrumental here in that document reclassification could be done off-line in a practical way as small changes in the classification

occur. One problem to be resolved in a multiple-user situation is whether all these users have the same view of the data base, and whetner one classification would be appropriate for all users. Would each user require his own classification, and if so, how could all these classification structures be economically stored and processed? Further research on real-time information retrieval systems should address these questions.

Further work is definitely indicated to investigate the Bayesian distance criterion for document classification and retrieval. A systematic investigation should be made of the parameter variations within the revised sequential method, and the method should be applied to more diverse and larger data bases. As mentioned above, more study should indicate how the Bayesian distance criterion can be used for keyword selection and modification. Reference [24] has shown that secondary classification can systematically be defined, and promises to extend to multiple classification. More research is required in order to develop this extension, so the revised sequential algorithm can achieve its full potential for automatic document classification and retrieval in a real-time user environment.

APPENDIX A

KEYWORD SET GENERATION FOR SPIN - 11 CLASSES

The following keyword sets were automatically generated for eleven class SPIN data as described in Section 2.3.1. These keyword sets are in support of the experiments described in Section 4.3.4, and summarized in Table 4.9. The sets are labelled A through H, and in the description, let P_i denote the relative frequency of the keyword in documents of category C_i. For brevity, the sets will be described in semi-tabular form.

KEYWORD SET	MAX NO. OF CATEGORIES	ONE CATEGORY TEST	TWO CATEGORY TEST	THREE CATEGORY TEST	FOUR CATEGORY TEST
A (481 words)	3	At least 2 documents, at least 3 occurrences	$P_1 > 0.60$	$P_1 > 0.50$ and $P_1 - P_2 \geq 0.20$	
B (437 words)	3	Subset of A found by removing words whose "good-bad" count satisfies BAD $\geq$ 2 + GOOD as determined in the experiment with keyword set A, d = 0.90, T=R=1.			
C (494 words) NOTE: $B \subseteq A \subseteq C$	4	At least 2 documents, at least 3 occurrences	$P_1 > 0.60$	$P_1 > 0.50$ and $P_1 - P_2 \geq 0.20$	$P_1 \geq 0.60$ and $P_1 - P_2 \geq 0.15$ and $P_2 - P_3 \geq 0.10$
D (534 words)	4	At least 2 documents, at least 3 occurrences	$P_1 > 0.60$	$P_1 > 0.50$ and $P_1 - P_2 \geq 0.20$	$P_1 \geq 0.50$ or $P_1 + P_2 \geq 0.75$
E (433 words) NOTE: $E \subseteq D$	4	Subset of D formed by removing words whose "good-bad" count satisfies BAD $\geq$ 2 + GOOD as determined in the experiment with keyword set D, α = 0.50, T=R=1.			
F (661 words)	4	At least 2 documents, at least 3 occurrences	$P_1 > 0.55$	$P_1 > 0.45$ and $P_1 - P_2 \geq 0.15$	$P_1 \geq 0.45$ or $P_1 + P_2 \geq 0.60$
G (620 words)	4	Subset of F formed by removing words whose "good-bad" count satisfies BAD $\geq$ 3 + GOOD as determined in the experiment with keyword set F, α = 0.70, T=R=1.			
H (571 words) NOTE: $H \subseteq G \subseteq F$	4	Subset of G formed by removing words whose "good-bad" count satisfies BAD $\geq$ 3 + GOOD as determined in the experiment with keyword set G, α = 0.50, T=R=1.			

APPENDIX B

DESCRIPTION OF ROCCHIO'S ALGORITHM

Rocchio's algorithm which has been referenced in Chapter 2 will be described here in greater detail.

ROCCHIO'S ALGORITHM

i) Choose any document d_k not yet assigned to a cluster and examine its similarity coefficients with the remaining unassigned documents;

ii) there must be at least N_1 documents whose similarity coefficients with d_k are at least p_1, and N_2 documents for which similarity coefficients with d_k exceed p_2;

iii) there must be between N_{min} and N_{max} unassigned documents whose similarity coefficients exceed p_x;

iv) if criteria (ii) and (iii) are satisfied, then this document together with all unassigned documents described by (iii) form a cluster;

v) repeat steps (i) and (iv) until all documents have been considered;

vi) if any document is still unassigned then recompute its correlation with each cluster as follows:

$$p(C_i,k) = \frac{1}{|C_i|} \sum_{j \varepsilon C_i} S_{jk}$$

where C_i is the cluster under consideration; $|C_i|$ is the number of elements in the cluster, and k is the unassigned document. p_1, p_2, p_x, N_1, N_2 are parameters which can be changed to vary the number of clusters obtained.

APPENDIX C

MEASURE OF COMPARISON OF TWO CLASSIFICATIONS

In Section 2.2.3, the problem was posed to measure the difference between two classifications K_1 and K_2 of the same set D, where either classification may contain overlapping categories. We postulated a measure, $||K_1 - K_2||$, as the minimum number of elementary changes required to transform K_1 into K_2. We defined

$(t_j)_i$ = the number of times the i^{th} object occurs in the j^{th} classification.

The $k_1 \times k_2$ matrix B:

$$b_{ij} = |C_i \cap C_j|,$$

where C_i is a category from K_1

C_j is a category from K_2

and where there are k_1 categories in K_1, k_2 in K_2.

The following theorem was postulated.

Theorem

If k is the maximum assignment sum for the matrix B, then

$$||K_1 - K_2|| = \{ \sum_{i=1}^{m} \text{Max}[(t_1)_i, (t_2)_i] \} - k.$$

Proof

First observe that an occurrence of an object need be involved in at most one elementary change in transforming classification K_1 to K_2. The maximum number of occurrences to consider creating, deleting, or moving from one category to another is $\text{Max}[(t_1)_i, (t_2)_i]$. The number of occurrences will be directly reduced by the number of elements in the intersection of corresponding categories for the category assignment that is made. Therefore, choosing the assignment so as to maximize the total number of such intersection elements will yield a minimum number of elementary changes required to change K_1 into K_2.

APPENDIX D

OPERATIONAL CHARACTERISTICS OF THE SEQUENTIAL ALGORITHM

The sequential algorithm was implemented in PL/I and executed on an IBM 370/165 computer. The classify program requires the following resources:

- A PL/I Level F compiler
- 252K bytes of main storage
- sufficient direct access storage for the frequency table
- sufficient secondary storage for the data base.

The execution time depends on the average number of keywords processed per document, the number and type of document input, and the T, R, and α values. Table D.1 lists CPU execution times for the sequential algorithm on the Kucera-Francis, Epilepsy, and SPIN data bases. The compile time for this classify program averaged 21 seconds.

Figure D.1 presents an overview of the CLASSIFY program, which is defined as SUPER-7. The following description indicates the function of each model.

Data Base	Number of Documents	Ave. No. of Keywords Per Document	CPU Time (Sec) (Go-Step only)	Type of Input
Kucera-Francis	326	6.7	29.42	Number format
Kucera-Francis	326	10.3	31.05	Number format
Kucera-Francis	70	8.0	14.56	Number format
Epilepsy	500	7.8	33.66	Number format
Epilepsy	500	4.9	23.88	Number format
SPIN	500	1.2	14.17	Full text
SPIN	500	1.6	12.70	Full text

Table D.1 Classify Program CPU Times

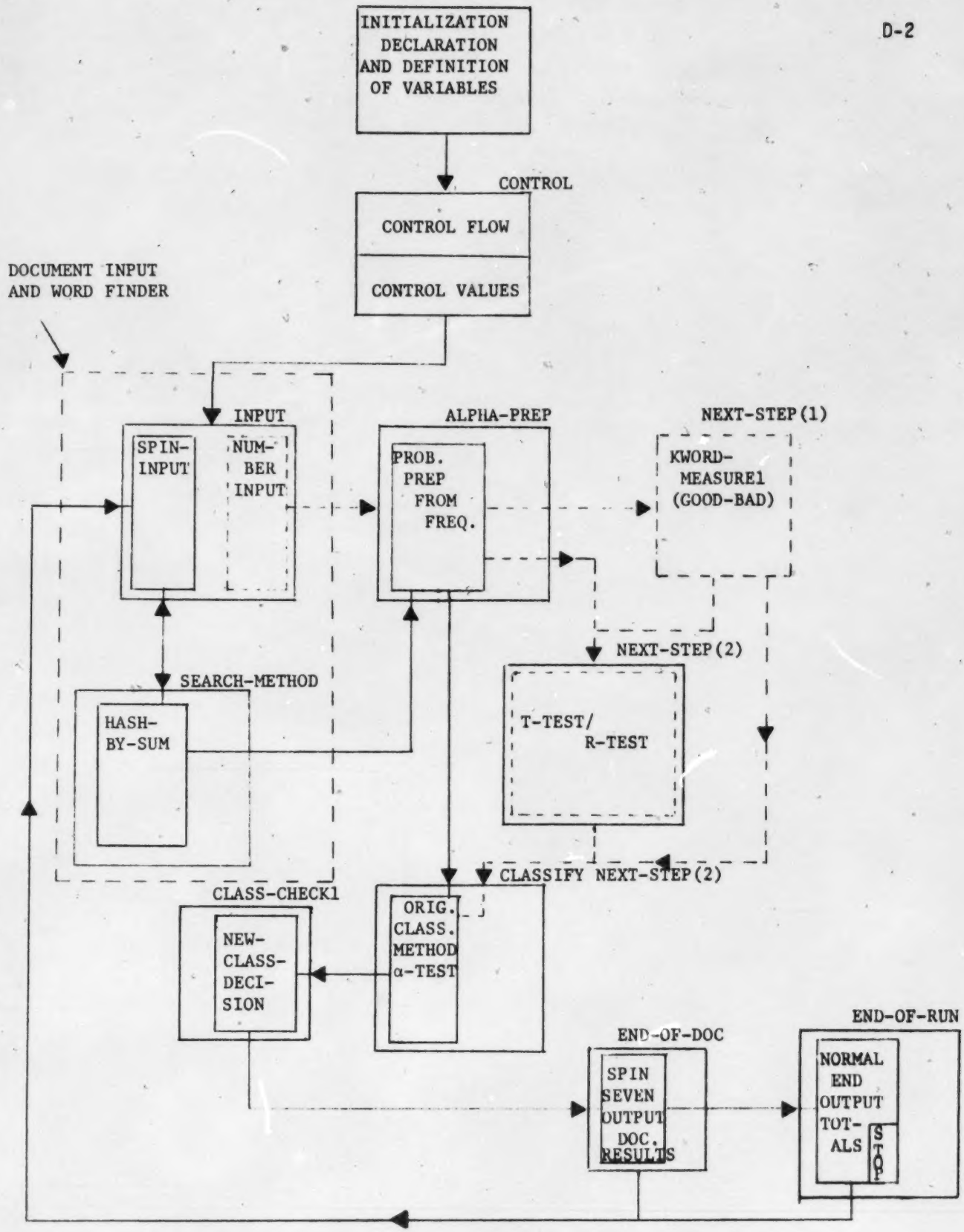
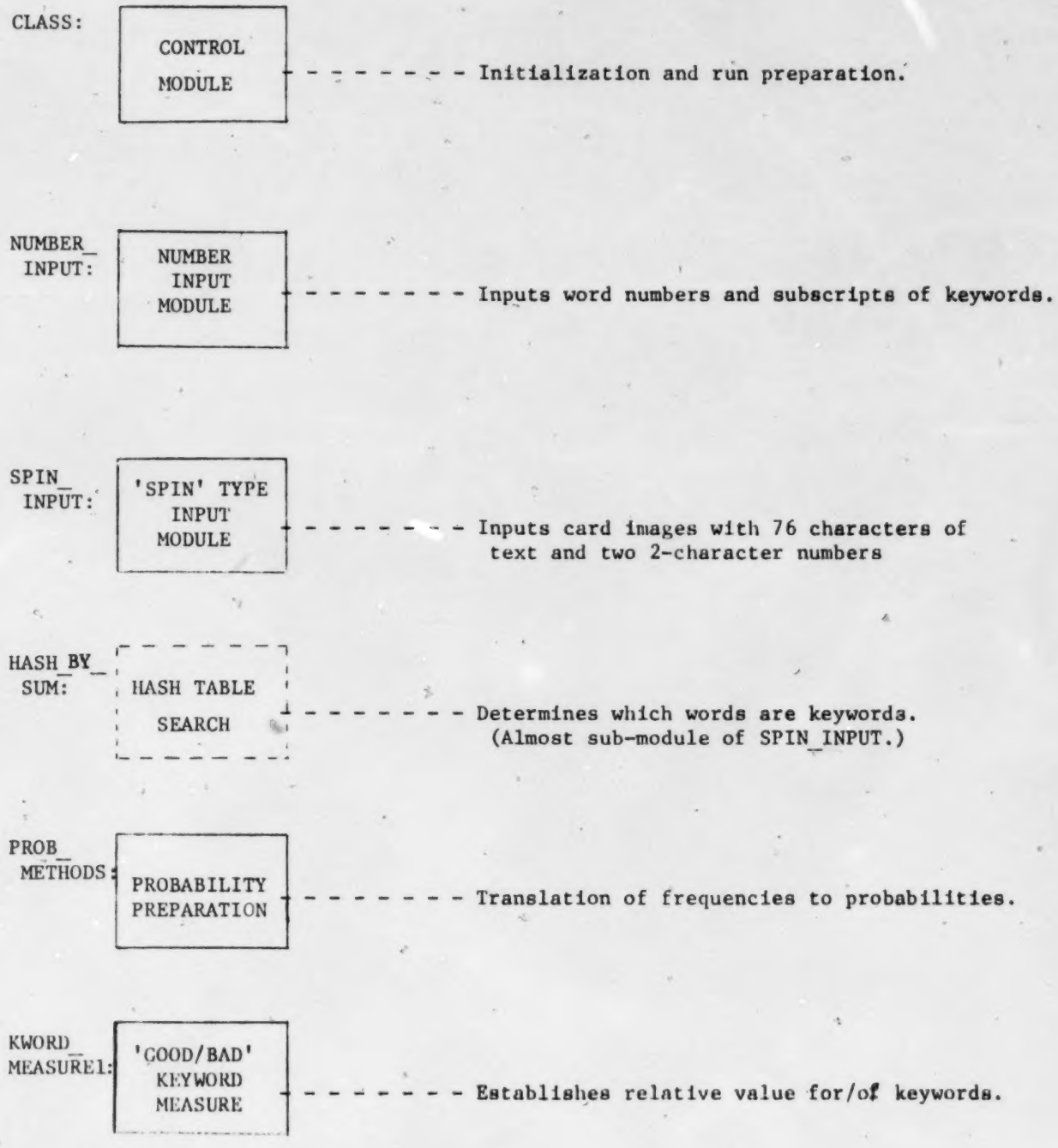
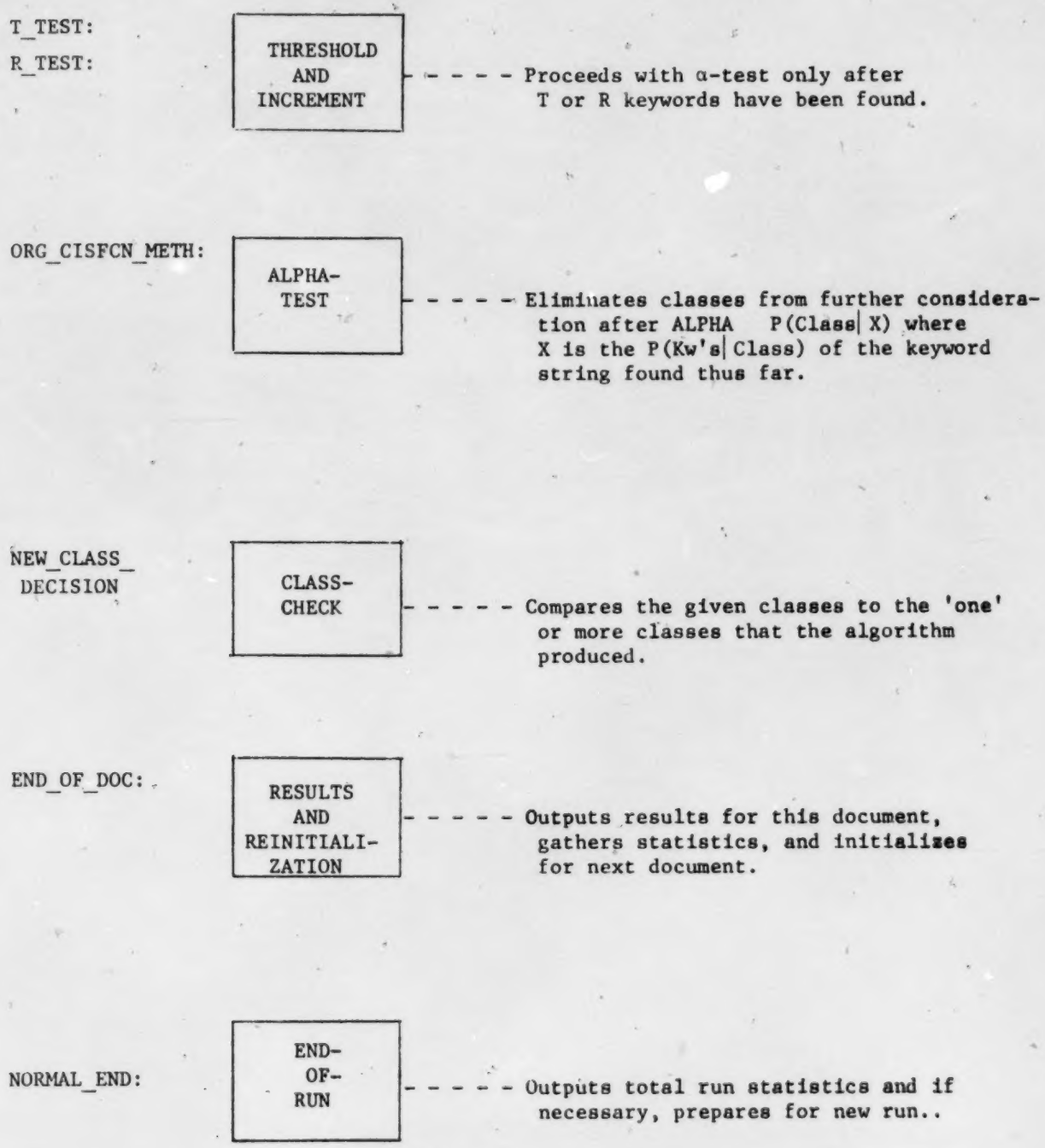


Figure D.1 Overview of CLASSIFY Program

Version DEW1 of CLASSIFY on IRCC83 DSN = FRJ15Ø.SUPERØ7

Label:




T_TEST:
R_TEST:
THRESHOLD AND INCREMENT
Proceeds with α-test only after T or R keywords have been found.
ORG_CISFCN_METH:
ALPHA-TEST
Eliminates classes from further consideration after ALPHA P(Class| X) where X is the P(Kw's| Class) of the keyword string found thus far.
NEW_CLASS_ DECISION
CLASS-CHECK
Compares the given classes to the 'one' or more classes that the algorithm produced.
END_OF_DOC:
RESULTS AND REINITIALI-ZATION
Outputs results for this document, gathers statistics, and initializes for next document.
NORMAL_END:
END-OF-RUN
Outputs total run statistics and if necessary, prepares for new run..

Summary of Functions

1. CONTROL MODULE

   a. Variable declaration and initialization

   b. Data input:

      This is accomplished by specifying a "change" number and other information as specified in the comments of the program. Some possibilities are as follows:

      Change #1) Inputs desired values of ALPHA, T, R for up to 20 individual passes over the data base.

      Change #2) Inputs the desired number of cells of the keyword hash table.

      ⋮

      Change #5) Inputs the keywords and hashes.

      Change #6) Inputs the keywords seqeuntially.

      ⋮

      Change #13) Sets a flag to skip default stoplist input.

2. NUMBER INPUT MODULE

   a. Inputs a number-of-words number then a keyword subscript.

   b. Based on an F(4,0) format and card images, 20 numbers are inputted at a time, 10 of each.

   c. FLAG(8) = '0'B means that the entire document has been read while FLAG(1) = '0'B means that all but one class has been eliminated. (The end of the document signal is a negative subscript.)

   d. Modifications exist to change encountered errors such as Line 110200 which should be LAST_POS = INDXX +1;

3. SPIN INPUT MODULE

   a. As 2) above, the classes that are given are translated. FLAGS (1-8) mean the same also.

   b. The text is inputted in this module, one record at a time.

   c. The output is one word.

4. HASH TABLE MODULE

   a. Input is a word.

   b. First the stoplist is searched, then the keywords are searched.

   c. Action is to go back to INPUT MODULE if word found in the stoplist or not found at all, or proceed to the next processing module with the subscript of the keyword and frequencies as its output.

5. PROBABILITY PREPARATION MODULE

   a. Inputs are the subscript and frequencies.

   b. Action: translates the frequencies to probabilities by dividing each positive frequency by the total frequency attributed that class. For those classes having no frequencies a default probability which is small but greater than zero is assigned.

   c. Output is $P(KW|C_i)$ where i = 0 to the number of classes minus one.

6. 'GOOD/BAD' MODULE

   a. Compares the P(KW|Classes given) to the P(KW|classes other) and accumulates values accorded that KW. Note: the only changes that can occur are in which classes are given as the probabilities are static.

   b. Modifications have been made to this module. Additions are also made to pass through this module and output the results.

7. THRESHOLD AND INCREMENT

   a. This module determines when the Bayesian rule is to be applied in what is normally called the ALPHA-TEST.

   b. Since the $P(X|C_i)$, X a keyword string, is calculated in 5 bits, T & R set too high generates underflow problems, or document read entirely problems for low keyword density documents.

8. ALPHA-TEST MODULE

   a. If a class, C_i, falls below α with its $P(C_i|X)$, X a keyword string, the class C_i is eliminated from further consideration. The calculation of $P(C_i|X)$ is:

$$P(C_i|X) = \frac{P(X|C_i)\ P(C_i)}{\sum_j P(X|C_j)P(C_j)}$$

   or Bayes' Rule.

   b. Certain modifications have been made to reflect the proper Bayes Rule above.

9. CLASS-CHECK MODULE

   a. This module has two parts: 1) the sole remaining class is compared to the given classes, and 2) if the document has been "read" entirely the remaining classes are compared to the given classes.

   b. Presently a document is considered "right" only when the remaining classes are a proper subset of the given classes.

10. END-OF-DOCUMENT MODULE

   a. Processing passes here after classification and the results are tabulated...

   b. Reinitialization ...

   c. Checking to see if pass over data base is through.

11. END-OF-RUN MODULE

a. Checks to see if more runs and if so prepares for the next ones.

b. Terminates otherwise after outputting the run statistics.

c. Modifications have been made to print the 'GOOD/BAD' results, etc.

APPENDIX E

KEYWORD HASHING ALGORITHM

The keyword hashing algorithm is described in terms of the following features:

a) table size: each hash table consists of a prime number of cells, a number of the form 4r + 3;

b) key generation: the key consists of the first eight characters of the given word, padded on the right with blanks, if necessary;

c) numerical transformation: sum the internal binary representation of the key, two characters at a time; this corresponds to adding four positive 16-bit binary numbers as the machine code is EBCDIC;

d) location address: the integer calculated in c) modulo the table size provides the entry point into the table;

e) collision resolution: the method used to resolve collisions is that developed by Day [10], which is an effective quadratic technique; this is modified in that a maximum chain length restriction is imposed when the tables are constructed or altered;

f) cell in the stoplist: each cell of the stoplist scatter table consists of ten bytes, corresponding to the first ten characters of the stoplist word to be hashed to that location; ten characters were found to be adequate to distinguish between all stoplist words;

g) cell in the keyword list: each cell of the keyword table consists of 14 bytes (corresponding to the first 14 characters of the keyword) and one byte for each class, corresponding to the frequency count of the keyword within that class.

It should be emphasized that in g) the probability $P(W_i|C_j)$ is <u>not</u> stored in fixed point decimal, as that would require one word of storage; instead, by storing the integer frequency, a considerable savings is achieved.

REFERENCES

1. Aberi, H. (1970) Implementation of the Fried Model of Automatic Classification, M.S. Thesis, The Ohio State University, Columbus, Ohio.

2. Augustson, J. G. and Minker, J. (1970) An Analysis of Some Graph Theoretical Cluster Techniques, *Journal ACM*, 17, 4, 571-588.

3. Baker, F. B. (1962) Information Retrieval Based Upon Latent Class Analysis, *Journal ACM*, 9, 4, 512-521.

4. Bonner, R. E. (1964) On Some Clustering Techniques, *IBM Journal of Research and Development*, 8, 1, 22-32.

5. Borko, H. and Bernik, M. (1963-64) Automatic Document Classification, Part 1: *Journal ACM*, 11, 2, 138-151.

6. Borko, H. (1968) Indexing and Classification, in H. Borko (ed.), *Automatic Language Processing*, John Wiley and Sons, Inc., New York.

7. Borko, H. (1962), The Construction of an Empirically Based Mathematically Derived Classification System, *Proc. 1962 Spring Joint Computer Conference*, 21, 279-289.

8. Cormack, R. M. (1971), A Review of Classification, *Journal of the Royal Statistical Society, Series A*, 134, 3, 321-353.

9. Dattola, R. T. (1968) A Fast Algorithm for Automatic Classification, Report ISR 14, Section V, Department of Computer Science, Cornell University, Ithaca, New York.

10. Day, A. (1970) Full Table Quadratic Searching for Scatter Storage, *Comm. ACM*, 13, 8, 481-482.

11. Doyle, L. B. (1965) Is Automatic Classification a Reasonable Application of Statistical Analysis of Text?, *Journal ACM*, 12, 4, 473-489.

12. Enslein, K. and Rose, C. L. (1970) Computer Based Data Analysis in Longevity Research, *Computers and Biomedical Research*, 4, 289-329.

13. Fairthorne, R. A. (1961) *Towards Information Retrieval*, Butterworth, London.

14. Forsythe, G. and Moler, C. (1967) *Computer Solution of Linear Algebraic Systems*, Prentice-Hall, Inc.

15. Foskett, D. J. (1963) *Classification and Indexing in the Social Sciences*, Butterworth, London.

16. Fried, J. B. and Landry, B. C., et al., (1968) Index Simulation Feasibility and automatic document classification, Technical Report 68-4, Computer and Information Science Research Center, The Ohio State University, NSF Grant No. GN-534.

17. Fu, K. S. (1968) *Sequential Methods in Pattern Recognition and Machine Learning*, Academic Press, New York.

18. Heaps, H. S. (1974) A Theory of Relevance for Automatic Document Classification, *Information and Control*, 22, 268-278.

19. Hillman, D. J. (1965) Mathematical Classification Techniques for Nonstatic Document Collections, With Particular Reference to the Problem of Relevance, *Classification Research, Proc. Second International Study Conference*, 177-209.

20. Hillman, D. J. (1964) The Notion of Relevance (I), *American Documentation*, 15, 1, 26-34.

21. Jackson, D. M. and White, L. J. (1971) The Weakening of Taxonomic Inferences by Homological Errors, *Mathematical Biosciences*, 10, 1, 83-89.

22. Jackson, D. M. and White L. J. (1972) Stability Problems in Non-Statistical Classification Theory, *Computer Journal*, 15, 3, 214-221.

23. Jardine, N. and Sibson, R. (1971) *Mathematical Taxonomy*, John Wiley and Sons, New York.

24. Kar, B. G. and White, L. J. (1975) A Distance Measure for Automatic Sequential Document Classification, Technical Report 75-7, Computer and Information Science Research Center, The Ohio State University, NSF Grant GN-36340.

25. Kucera, H. and Francis, W. N. (1967) *Computational Analysis of Present-Day American English*, Brown University, Providence, Rhode Island.

26. Luhn, H. P. (1957) A Statistical Approach to Mechanized Encoding and Searching of Literary Information, *IBM Journal of Research and Development*, 1, 4, 309-317.

27. Maron, M. E. (1961) Automatic Indexing: An Experimental Inquiry, *Journal ACM*, 8, 3, 407-414.

28. Mathis, B. and White, L. J. et al., (1970) Stability Analysis of Term Similarities for Information Classification, Technical Report 70-4, Computer and Information Science Research Center, The Ohio State University.

29. Rocchio, J. J. (1966) Document Retrieval Systems - Optimization and Evaluation, Report No. ISR 10 to the NSF, Department of Computer Science, Cornell University, Ithaca, New York.

30. Rush, J. E. and Salvador, R. et al. (1971) Automatic Abstracting and Indexing, *Journal American Society for Information Science*, 22, 4, 260-274.

31. Rush, J. E. and White, L. J. (1973) An Automatic Classification System With Application to Various Information Retrieval Environments, Department of Computer and Information Science, The Ohio State University, Quarterly Report to NSF, Office of Science Information Service on research grant No. GN-36340.

32. Salton, G. (1968) *Automatic Information Organization and Retrieval*, Chapter 4, McGraw-Hill Book Company, New York.

33. Salton, G. (1965) The Evaluation of Automatic Retrieval Procedures - Selected Test Results Using the SMART System, *American Documentation*, 16, 3, July 1965.

34. Salton, G. (1971) *SMART Information Retrieval System*, Prentice-Hall, Englewood Cliffs, New Jersey.

35. Searchable Physics Information Notices (SPIN) - Technical Specification (1971) AIP ID 71S, American Institute of Physics, New York.

36. Sokal, R. R. and Sneath, P. H. A. (1963) *Principles of Numerical Taxonomy*, Freeman, San Francisco.

37. United States Patent Office (1970) Computer Aided Classification, Department of Commerce, Washington, D.C.

38. Van Rijsbergen, C. J. (1971) An Algorithm for Information Structuring Retrieval, *Computer Journal*, 14, 4, 407-412.

39. Vickery, B. C. (1959) *Classification and Indexing in Science*, Butterworth, London.

40. Vickery, B. C. (1965) *On Retrieval System Theory*, Butterworth, London.

41. Wald, A. (1947) *Sequential Analysis*, Chapter 10, John Wiley and Sons, Inc., New York.

42. Williams, J. H. Jr. (1963) A Discriminant Method for Automatically Classifying Documents, *Proc. 1963 Fall Joint Computer Conference*, 24, 161-167.

43. Williams, J. H. Jr., (1965) Results of Classifying Documents with Multiple Discriminant Functions, *Proc. Symposium of Statistical Association: Methods for Mechanized Documentation*, Washington, D. C., 1964 NBS Publication 269, 217-224.

44. Yu, C. T. (1974) A Cluster Algorithm Based on User Queries, *Journal American Society for Information Science*, 25, 4, 218-226.